Multi-Object modelling of the face

Hanan Salam

N° d'ordre : 2013-35-TH

SUPELEC

Ecole Doctorale MATISSE

« Mathématiques, Télécommunications, Informatique, Signal, Systèmes Electroniques »

THÈSE DE DOCTORAT

DOMAINE : STIC
Spécialité : Traitement du signal et telecommunication

Soutenue le 20/12/2013

par :

Hanan SALAM

Modélisation Multi-Objet du visage

Directeur de thèse :	Renaud Seguier	Professeur (IETR)
Composition du jury :		
Président du jury :	Lionel PREVOST	Professeur (LAMIA)
Rapporteurs :	Saidi BOUAKAZ	Professeur (LIRIS)
	Jean-Claude MARTIN	Professeur (LIMSI)
Examinateurs :	Piere-Yves COULON	Professeur (Gipsa-Lab)

Abstract

The work in this thesis deals with the problem of face modeling for the purpose of facial analysis. Facial modeling was used to achieve gaze and blink detection and emotion recognition.

In the first part of this thesis, we proposed the Multi-Object Facial Actions Active Appearance Model. The specificity of the proposed model is that different parts of the face are treated as separate objects and eye movements (gaze and blink) are extrinsically parameterized. Starting from a learning database that contains no variations in gaze or blink, the model is able to follow the movements of the eyeball and eyelids, which increases the robustness of active appearance models (AAM) by restricting the amount of variation in the learning base.

The second part of the thesis concerns the use of face modeling in the context of expression and emotion recognition. First we have proposed a system for the recognition of facial expressions in the form of Action Units (AU). The proposed system is based on the combination of Local Gabor Binary Pattern Histograms (appearance features) and AAMs (hybrid features: appearance and geometry) using Multi-Kernel Support Machine Vectors. Our contribution concerned mainly the extraction of AAM features. The AUs to detect concerned the upper and lower part of the face. Thus, we have opted for the use of local models to extract these features. Results have demonstrated that the combination of AAM with the LGBP appearance features has led to ameliorate the results of recognition. This system was evaluated in FERA 2011, an international challenge for emotion recognition of which our team have took the first place.

The second system concerns the multi-modal recognition of four continuously valued affective dimensions: arousal, valence, power and expectancy. We have proposed a system that fuses audio, context and visual features and gives as output the four emotional dimensions. The visual features are in the form of facial expressions. More precisely, we have found that the smile is a relevant cue for the detection of the aforementioned dimensions. To detect this feature, AAM is used to delineate the face. We contribute at this stage of the system to find the precise localization of the facial features. Accordingly, we propose the Multi-Local AAM. This model combines extrinsically a global model of the face and a local one of the mouth through the computation of projection errors on the same global AAM. The proposed system was evaluated in the context of AVEC 2012 challenge and our team got the second place with very close results to those who came in the first place.

To my parents...The source of my pride

إِلَى أُمِّي وَ أَبِي...

لَولَاكُمَا لَمَا وَصَلتُ إِلَى مَا أَنَا عَلَيه.انَّنِي فَخُورَةُ بِكَونِي إِبنَتُكُمَا اكثَر مِمَّا انتُمَا فَخُورَان بِي

Acknowledgments

The work in this thesis is the result of my research and exchange with several persons during the simultaneously long and short past three years in Supelec, team SCEE, IETR. These years were tremendously enriching together at the professional and at the personal levels. The first person I would like to thank is my thesis supervisor Renaud SEGUIER. He has guided me through the work of this thesis, given me advice and encouragements to continue. I am really grateful for his moral support, for not stressing me at all, for being patient and understanding especially when I had so much questions, and for making me believe more in myself and my work. Working with him has enriched me by lessons about success, positive thinking, taking risks, moving forward and enthusiasm. It is a great pleasure to work with such a person. I personally think he is the best supervisor anyone can ever have.

I am also thankful for Christophe MOY for being the director of my thesis for two years and for his readiness to help me through the work of this thesis. His encouragement and advice have helped me a lot especially his advice about the writing of this repport. I am really grateful for his kindness and support and it was a pleasure to be directed by him.

I also want to thank Nicolas STOIBER for the several scientific meetings of which we have exchanged ideas. I am thankful to him for responding to my questions and for his valuable remarks that have been a source for advancing this work.

I would also like to pay my sincere gratitude to Jacques Palicot, the director of the team SCEE. His advice has helped a lot especially before my final presentation.

I think SCEE is a really comfortable place to work in because of its friendly work environment. Everybody is welcoming and friendly. I thank professors Yves LOUET, Daniel LE GUENEC and Christophe MOY for being friendly and welcoming. I thank the administration of Supelec for being very helpful and kind. I equally thank the 5050 team of Supélec for responding fast to the several technical obstacles that my computer went through during this thesis.

I would like to thank Professor Saida BOUAKAZ, Université Claude Bernard Lyon1

(LIRIS) and Jean-Claude MARTIN, Université Paris Sud (LIMSI)for accepting to be the rapporters of my thesis. I also want to thank the other members of the jury Professors Lionel PREVOST, Université des Antilles et de la Guyane (LAMIA), and Pierre-Yves COULON, Grenoble-INP (Gipsa-Lab) for being present to judge my research work in this thesis.

I am really thankful to the staff of Dynamixyz. Thanks to Gaspard Breton for permitting me to work at Dynamixyz when Supelec was closed and thanks to all the others at Dynamixyz for welcoming me in their workplace.

I want to say thank you to my friends and colleagues Ziad, Oussama, Caroline, Xi guang, Lamarana, Abel, Samba, Marwa, Jerôme, Catherine, Salma, Patricia, Wassim, Vincent. They have all supported me during hard periods of the thesis. Thank you Caroline and Marwa for being so supportive in some hard times. Thank you both for being "Study Partners" in the "No-Motivation" days. A big thanks goes also to my friend Oussama for making the work place more fun. The breaks we did were very helpful to work more efficiently.

A special thanks goes to my compatriots, friends and sometimes roommates: Farah, Riham, Lama and Hussien. Without them it would have been so much harder to go through these years. I am thankful for their support and for being there to fill out the leisure time. I would not forget Nour Soleil, my friend since the first year of university. She has massively supported and encouraged me to work especially during the period of writing my thesis.

A particular thanks goes to Professor Mohamed ZOAETER, the former dean of the faculty of engineering at the Lebanese university, for opening the doors between the Lebanese university and the universities of France. In my opinion, his diligence has a lot contributed in the advancement of the research in Lebanon.

The warmest gratitude goes to my beloved parents, Ali and Salma, for their continuous emotional support. Thank you father for letting me pursue my ambitions with so much love and support. Thank you mother for your kindliness and prayers. I thank my four brothers for their humor which made the journey of my studies lighter and more amusing even if it was from a distance. Without my family my success would not have been possible.

Hanan SALAM

Contents

List of Figures

List of Tables

Introduction

> *The human face – in repose and in movement, at the moment of death as in life, in silence and in speech, when seen or sensed from within, in actuality or as represented in art or recorded by the camera – is a commanding, complicated, and at times confusing source of information.*
>
> – P. Ekman, W. Friesen, and P. Ellsworth, 1972, p.1

Context and Motivation

Human-Human Interaction

Humans usually communicate with each others through two forms: verbal and non-verbal communications. Verbal communication is communicating through spoken words whereas non-verbal communication is communicating through exchanging visual and vocal wordless cues. Figure 1 illustrates the different cues of non-verbal communication. These cues can be face-related (facial expression, head pose and eye contact), speech-related (volume, pitch, tonality, etc...) or body-related (gestures and touch, body language or posture, etc...).

Albert Mehrabian [MW67], a pioneer researcher of body language, has stated that in a face-to-face communication, 55% of human communication is visual whereas 38% is vocal (speech tone, pitch, volume...) and only 7% is verbal. The face, therefore, can be considered as the most powerful cue of nonverbal communication. Very important messages are encoded in our facial expressions, and during our daily lives, we concurrently decode the facial messages encoded in the faces of others. In the simplest interaction, the face is the gravity of our attention. We analyze it to read information that gives us clues about the person's identity, age, emotions, intentions, attraction and even personality.

"*The Eyes are the window to the soul*", a very famous English proverb, demonstrates one very important feature of the face: the eye. This feature with its actions that can be grouped into saccades, fixations, blinking and winking carry information about the person's intentions, thinking and interior emotions. Moreover, the eyes language is known among all cultures where people communicate with their eyes to send messages to each

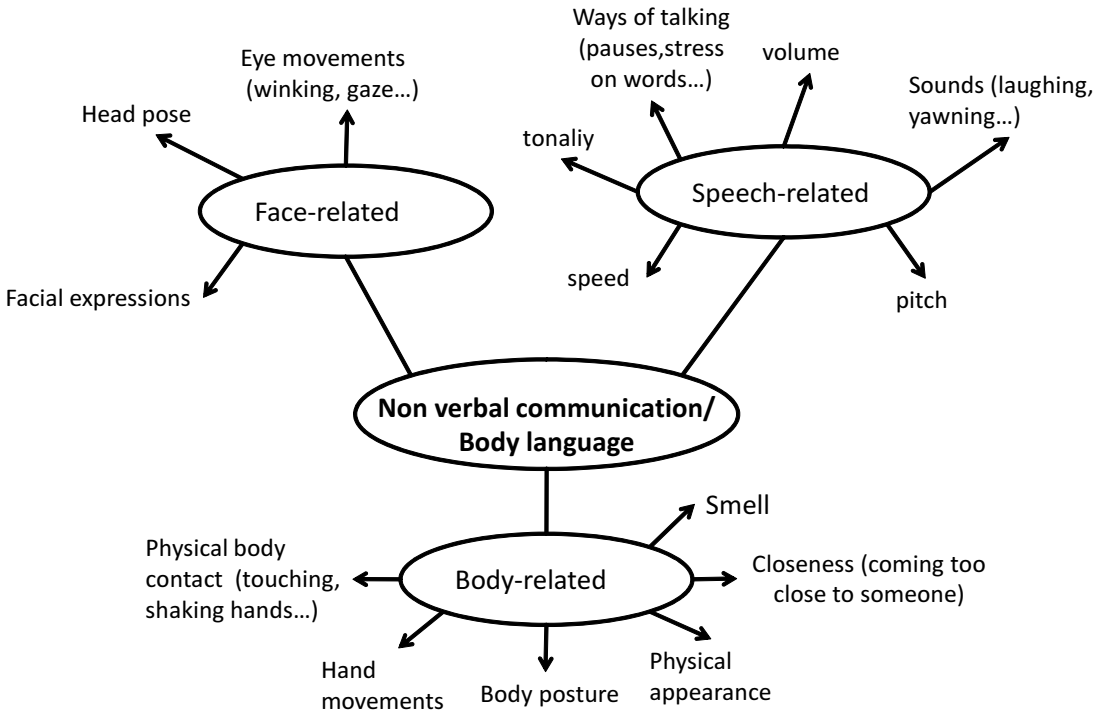


Figure 1: The different cues of non-verbal communication

other. For instance, staring might mean attraction and rolling eyes means dislike of the other's speech.

Human-Computer Interaction

> *Indeed the way that people use machines is of key importance. The most significant advances in computer science will be those facilitating this interaction.*
>
> – T. Winograd and F. Flores, Understanding Computers and Cognition 1986, p. 1371

With time, the need for Human Computer Interaction (HCI) is increasing more and more. Computers had passed from being a completion to ones life to being a necessity. In addition, HCI had grouped multi-disciplinary fields ranging from psychological and medical to entertainment fields.

For a long time, interacting with computers was done through the mouse and the keyboard. However, this does not seem enough. Interaction with computers happens now in our daily lives through the use of our laptops, our tactile phones and tablets and the

need for other ways of interaction is mandatory. A recent direction is the integration of face and eyes information as an alternative.

When it comes to Human-Human Interaction (HHI), the task of understanding others by analyzing their faces or eyes is straightforward. Our minds are used to such analysis. In contrary, when it comes to Human Computer Interaction (HCI), arriving at communicating with the computer in a similar manner as in HHI is indeed difficult. The ultimate dream is a human-computer interaction that resembles a human-human interaction. That is, a multi-modal verbal and non verbal communication.

This dream carries us towards the field of automatic face processing. The latter consists of two directions: automatic facial analysis and facial synthesis. In order to arrive at interacting with computers in a similar way to interacting with humans, computers should be able to analyze faces similar to the analysis of humans and to synthesize facial gestures and expressions the most realistically possible. This includes the ability of the computer to understand the person's intentions, emotions and behavior and reproduce them.

Problem Statement

Automatic face analysis

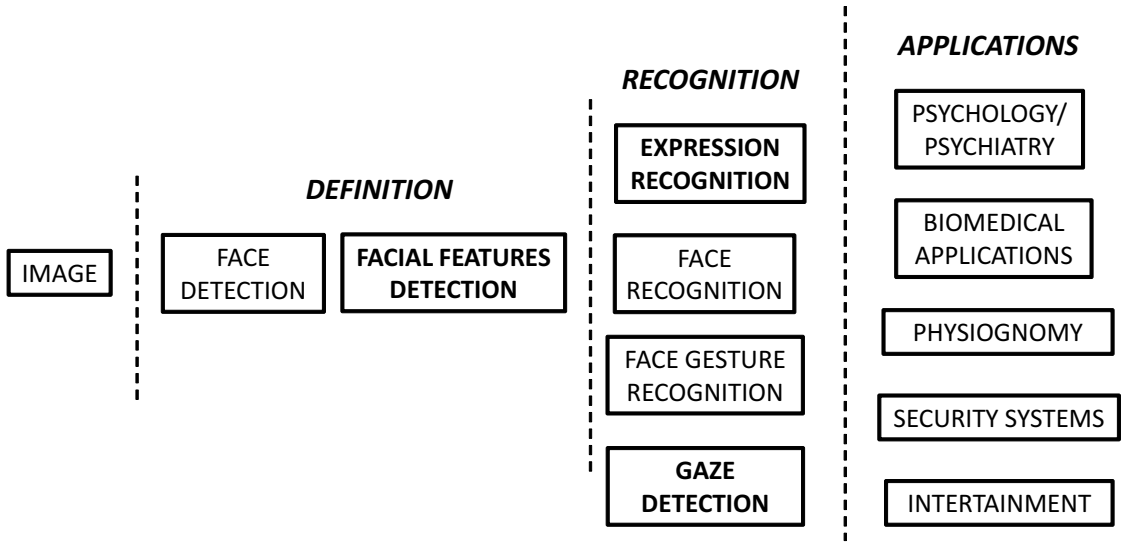


Figure 2: Automatic face analysis structure

The task of automatic face analysis is indeed a complex and difficult one. The reason is that individuals faces have different physiognomies. The person's age, gender, ethnicity, facial hair, make up, and occlusions due to hair and glasses, all play a significant role in establishing the difference in appearance among individuals. In addition, the variability

in head pose and illuminations conditions makes the task of automatic face analysis more and more difficult.

When we speak about automatic facial analysis, three levels are faced: The level of definition, the level of recognition and that of application. Figure 2 depicts these three levels.

- **The level of definition** and this includes face detection and facial features detection. Face detection is detecting the location of the face inside the image. Facial features detection can be done in two forms: detecting high level or low level information. The former is finding the precise locations of the traits of the face which includes the eyes, mouth, eyebrows and chin. The latter is detecting image information such as edges, color, or intensity.
- **The level of recognition** and this includes the mapping of the extracted information during facial definition to more concrete information. This information can be in the form of face recognition, expression recognition, facial gestures recognition or gaze detection.
- **The level of application**, this means the integration of the automatic face analysis systems in different domains of Human Computer Interaction applications. For example, these can be integrated in physiognomy (relationship between facial features and personality traits), security systems, clinical psychology and psychiatry, biomedical applications, lip reading to assist the speech recognition in noisy environments, low bit rate video coding of the face for telecommunication, avatar systems, and entertainment.

Among these three levels, this thesis main contributions are situated in the first two levels, that is the levels of definition and that of recognition. We are mainly concerned in the face modeling rubric and the use of this modeling for emotion and expression recognition tasks.

Face Modeling

In order to define the face and extract its features, a face model should be implemented. This model should be able to deal with the aforementioned constraints that the face is subjected to during the automatic face analysis. For instance, it should be robust to variabilities between the appearances of different faces, due to lighting conditions, imaging conditions, makeup, the presence or not of several factors such as eyeglasses, hair, mustache, beard, etc. The same face model should be able to account also for changes due to variable head pose orientation and facial actions including facial expressions.

Active Appearance Models (AAM) [CET98b] are statistical tools that are used to model the face shape and appearance. Starting from a set of learning examples, these models are able to find a set of parameters describing the face shape and appearance. Such models had proved to be very efficient in modeling faces. They belong to the class of Analysis-by-Synthesis models that find the optimal vector of parameters through the minimization of the difference between a synthesized image by the model and the real image. Such models

present drawbacks concerning the ability to generalize to new data that are not present in the learning database and parameters that are not well-defined.

Thesis objectives

The objective of this thesis is to make an advance in face modeling which can be used in face analysis in the context of HCI and automatic human interpretation. The implemented methods would facilitate multi-modal non-verbal communication between humans and computers.

The first part of this thesis concentrated on face modeling through the implementation of possible solutions to solve for the limitations of AAM. Such solutions have led to the conception of a system that is capable of analyzing eye motions, specifically eye gaze and blinking of which are known to be very important channels of non-verbal communication between humans.

The second part of this thesis have concentrated on the application of face modeling for expression and emotion recognition. For this we have participated in two major challenges in the context of a 3D immersion ANR project based on emotional interaction (IMMEMO): The first "Facial Expression Recognition and Analysis Challenge (FERA 2011)" in collaboration with the *Institut des Systèmes Intelligents et de Robotique* (ISIR), University of Pierre and Marie Curie and LAMIA, University of the West Indies and Guiana. This challenge is about detecting expressions in videos in the form of Action Units or discrete expressions. The second "International Audio / Visual Emotion Challenge and Workshop (AVEC 2012)" in collaboration with *Laboratoire Traitement et Communication de l'Information* (LTCI), Telecom ParisTech and the society Dynamixyz, Rennes. It presents a platform for combining different modalities, mainly audio and visual for the purpose of emotion recognition. Such combination resembles the real nature of non-verbal communication.

The purpose of these challenges is to advance expression and emotion recognition to process data with naturalistic behavior in large volumes that are not segmented or prototypical which is the type of data that HCI would face in the implementation of real applications. Another objective is providing a common database for researchers to compare their systems.

Thesis outline and contributions

The chapters of this thesis expose how we contribute to face modeling using Active Appearance Models and how we employ the proposed model into a gaze detection system. In addition, they show our contributions in two grand challenges for expression and multimodal emotion recognition.

As face modeling is a fundamental rubric for both facial analysis and facial synthesis, we present in chapter 1 a state of the art on both axes. Section 1.1 is dedicated to review the previous work in the analysis domain and section 1.2 reviews those in the synthesis

while focusing on facial deformation modeling and synthesis. Section 1.3 reviews methods that are based on an analysis-by-synthesis loop. A state of the art on the subjects of gaze and blink detection is included in this chapter in section 1.4.

In chapter 2, we present the theoretical background of the basic Active Appearance Models [CET98b] in addition to the difficulties and limitations encountered by this classical formulation.

Chapter 3 represents the first and main contribution of this thesis. We present a new active appearance face model that uses ideas from the computer graphics domain. It combines benefits of statistical models and interpretable parameterizations of geometric models. It deals with the face as an aggregation of separate objects. These objects are related to each other through a multi-objective optimization framework. The resulting model contributes to Active Appearance Models by restricting images in the facial database.

Chapter 4 presents our contributions in two recognition systems through our participation in two grand challenges: The Facial Expression Recognition and Analysis Challenge (FERA 2011) and the Audio/Visual Emotion Challenge (AVEC 2012).

Chapter 1

Face Modeling: A state of art

Sommaire

Modeling faces has took a great portion of research in the fields of Computer Graphics and Computer Vision. These two fields use face modeling, each in its own way. In Computer Graphics, face modeling is essential for facial synthesis and animation. In Computer Vision, modeling the face and its deformations serves at the automatic analysis of faces.

Even though these two fields intersect in the necessity of realistic modeling of the face, however they remain two separate fields that have their own techniques and associated difficulties. In this chapter, we explore the state-of-the-art methods in face modeling in both domains: Synthesis and Analysis of human faces. The reason is that we would like to explore the modeling techniques of both fields, make a link between them and see if we can bring ideas from one field to another.¡

Thus, we choose to classify the state-of-the-art in face modeling according to the field of application: Analysis, Synthesis and Analysis-by-Synthesis. The Analysis methods are the set of techniques that are used for the purpose of facial analysis, they are not able to synthesize faces or their deformations. The output is the detected facial landmarks representing the shape of the face in question. The synthesis methods are those that are used in the field of computer graphics for the purpose of creating and animating faces. Finally the Analysis-by-Synthesis are those that analyze the face through an analysis-by-synthesis loop. Such models are able to both analyze and synthesize faces. The following sections resume the different face models present in the literature and figure 1.1 depicts the flow chart of the classification of the different methods of the state of art.

In addition to face modeling, we review past literature in blink and gaze detection. We find this review necessary since in the work of this thesis we concentrate through face modeling on the actions performed by eyes due to their importance to non-verbal computer-to-face communication.

1.1 Analysis models

Analysis models are those that are meant to analyze the face in terms of its shape or texture. We classify them into three categories: Hand-crafted models, active contours and

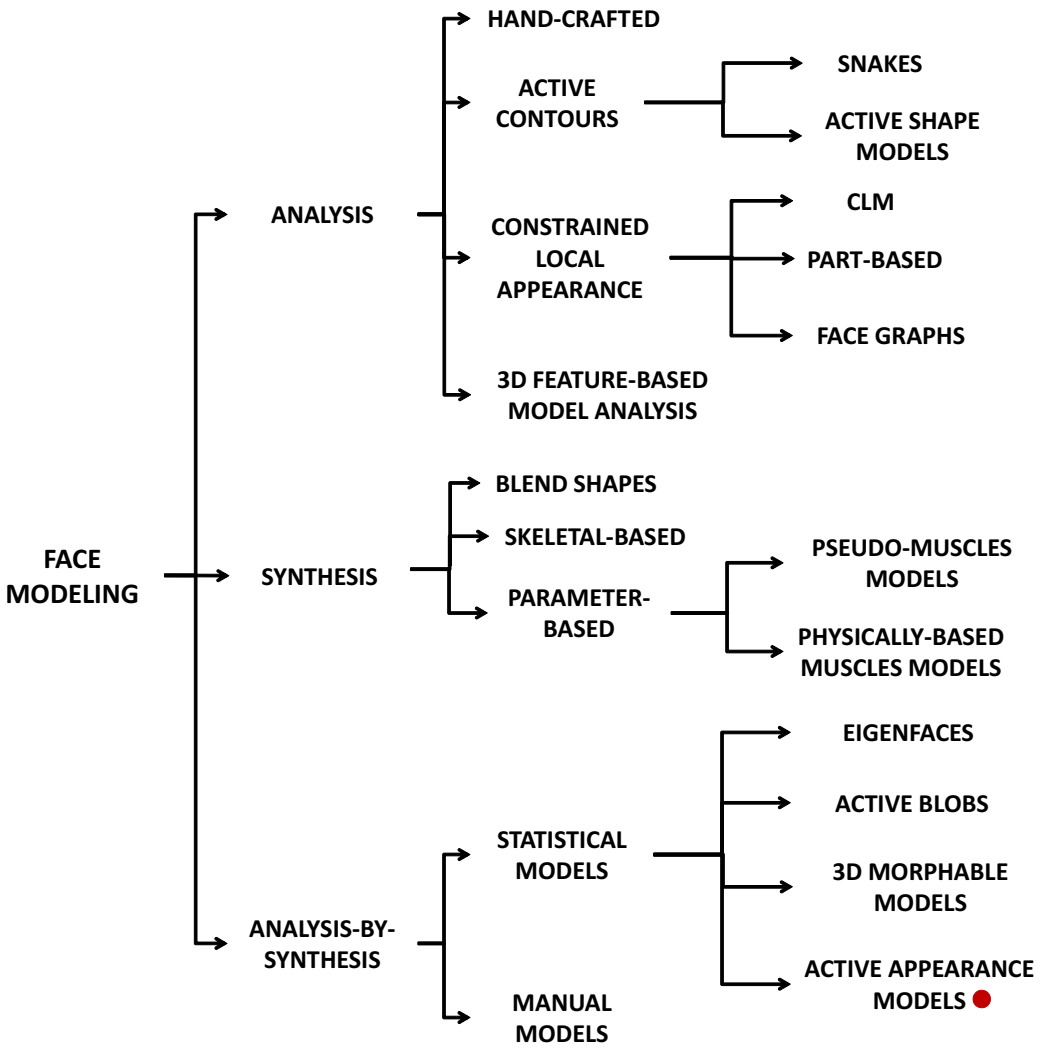


Figure 1.1: Flow chart of the state-of-the-art classification

constrained local appearance models. Hand-crafted methods manually design models for the different features of the face. Active contour methods are those that deform themselves to delineate the face. Constrained local appearance models are approaches that constrain local models of appearance by global models (sections 1.1.1 to 1.1.3). After obtaining the shape, some methods use it to fit a 3D model permitting to describe the face in terms of deformation parameters (section 1.1.4). The following sections resume these different methods and discuss their advantages and disadvantages.

1.1.1 Hand-crafted models (manual models)

The simplest approach one can think of to model the face is to manually design a model. [YHC92a] builds a parametrized deformable template model of the eye and the mouth. The eye model is formed by parabolic curves and circles. Two mouth model templates are designed: one for describing a closed mouth and one for an open mouth. Mouth models are also modeled by parabolas connected with each other. The templates are assigned energy functions that relate them to the image's low level information such as intensity, edges and valleys. [WLZ04] also uses eye templates for eyelids extraction. They remove the iris template from the eye template of [YHC92a] arguing that a template dedicated to eyelids only improves the eyelids localization permitting a better iris localization afterwards. In addition, they add two energy terms to overcome problems of shrinking and rotation of the template. [BQ06] improves the model of [YHC92a]. They propose to find the eyes, nose and mouth regions beforehand using an appearance template matching procedure. Deformable template of Yuille is then applied on these regions. Energy functions are designed based on edges, weighted mean and variance of the image's intensity. [MP00] interpolates B-spline curves between Facial Feature Points of the MPEG4 [EHRW98] head model to create a face template.

Even though such approach is effective, however, designing a model is a complicated task where each feature of the face should be modeled by a specific model. It is computationally demanding due to the number of parameters and the number of different energy functions associated with it. In addition, such methods may require high contrast images because they depend on the intensity of the image. They should be initialized near the object in question. Morever, they are not flexible enough to deal with variations in head pose. And, as we saw different templates should be designed to deal with different states of the object. For instance, two models are needed to model an open and a closed mouth. This is similar to modeling a closed eye where the same template is not able to deal with both open and closed eye.

1.1.2 Active contours

Active contours are deformable shapes that deform themselves to delineate the face. They include snakes that are also called active contours and Active Shape Models.

1.1.2.1 Snakes

As low level information of the image can lead to unsatisfactory results for tracking features, [KWT88] proposed the use of high level information in order to drive the output to the desired place. Snakes were the fruit of this proposition. A snake is an elastic, continuous flexible contour which is fitted to the image by an energy minimizing mechanism. Prior knowledge on the object to be tracked in the image is encoded in its elastic energy which relies on gray-level gradient information. This energy constitutes internal and external energies. The first is responsible for stretching and bending the snake (elastic

energy) which controls the geometry of the contour. The second is a sort of image forces that drive the snake towards salient image features such as lines and edges. It is minimal when the snake is at the object's boundary (controls image shape). According to the object in question, adequate energy functionals should be designed that take into account the object specificities. An excellent explanation of active contours can be found in the book of Blake and Isard [BI98].

Many extensions where proposed to ameliorate the snake algorithm. Among these, we cite the rubber snakes [RM95]. They incorporate the gradient orientation of image edge points through the construction of a Signed Distance Potential (SDP) of which they define the snake's external energy. This ensures a global deformation of the snake. Another extension is the no-edges approach of [CV01] where the snake's energy term is independent of the object's boundary. However, the snake is able to stop at the boundary of the object to be detected. [PCMC01] integrates region information through the introduction of a region energy term. This robustifies the results of search and makes it less sensitive to initialization of the snake. Geodisic active contours (GAC) is the result of combining active snakes with level-set methods and curve evolution theory [CKS97, GKRR01]. This type of snakes uses geometric measures thus making the snake less dependent to parameterizations. Higher dimensional snakes were also proposed. [Hsu02] introduced the 2.5D snakes which utilizes 3D shape coordinates instead of 2D, and the interacting snakes where multiple snakes corresponding to the different parts of the face are manipulated iteratively to minimize the attraction and repulsion energies of these snakes simultaneously. They interact with each other in order to adapt to facial components.

B-spline curves can be seen as a least-squares style snake [WH04]. Parametric B-spline curves were proposed by [BCZ93] to track contours. Control points of the B-spline curve were tracked so as to match the contour in question. [MSMM90] improves the speed of matching snakes by approximating the snake's curves using B-spline curves. Later, [XY09] increases the speed of B-spline snakes by replacing the external force by a gradient vector flow.

Snakes are efficient for features segmentation because they are autonomous and self-adapting in their search for a minimal energy state [P.97]. In addition, they can be easily manipulated using external image forces. However, their drawback is that they are too flexible to limit their deformations to a reasonable amount of variations for a specific object and they do not have the ability to specify a specific shape.

1.1.2.2 Active shape models

Active Shape Models (ASM) are one type of models that have the ability to restrict variations into a specific amount and to a specific object. They were first proposed by [CCTG92, CTCG95]. Such models take advantage of the point distribution model to restrict the shape range to an explicit domain learned from a training set. They differ from snakes in that they use explicit shape model to place global constraints on the generated shape. The ASM scheme works as follows: After annotating the facial images

present in the training set, Principal Component Analysis (PCA) models the relationships between the resulting set of points. To find the facial points, each landmark is localized independently by searching for the strongest edge along a profile normal to the model boundary and centered at the current position of the point(assuming that the boundary point is on the strongest edge). Many extensions were later proposed to ameliorate the performance of ASMs. [CT99] models the distribution of local appearances by a mixture of multivariate Gaussians, rather than a single Gaussian distribution used in the classical ASM. [ZGZ03] proposes the Bayes Tangent Shape Modes (BTSM). They project the shapes into a tangent space and then employ a Bayesian inference framework to estimate the shape and pose parameters. [RG06] improves the ASM search by applying a robust parameter estimation method using M-estimator and random sampling approaches to minimize the difference between the model shape and the suggested one. [CC07] use non-linear boosted features trained using GentleBoost instead of local profiles around the feature points. Their method improves the computational efficiency and tends to give more accurate results. [LI05] propose to build texture models with AdaBoosted histogram classifiers which bypasses the need to assume a Gaussian distribution of local appearances. They also propose a new shape parameters optimization method based on dimensional tables to restrict the shapes into allowable ones after proving that the original bounds used to constrain the shapes in ASM is not that efficient. The method improves robustness of landmark displacement, illumination, different facial expressions, specific traits such as mustaches, glasses, and occlusions. [MN08] propose the Extended ASM (EASM) which extends the ASM by increasing the number of landmarks such that fitting a landmark tends to help fitting other landmarks. They also use two instead of one-dimensional landmark templates, add noise to the training set and apply two consecutive Active Shape Models. The model works well with frontal views of upright faces but no results were claimed about robustness to factors such as head pose or lighting. To deal with cases where boundaries of the object are not clear, limited image resolution, or missing boundary information, [YXTK10] propose the Partial ASM (PASM) that selects salient contour points among the original points of the shape to estimate the shape. The salient feature points of the shape contour are detected using a normal vector profile (NVP) [HFDL06] method. Some authors employed hierarchical systems to robustify the ASM results. For instance, [TBA+09] combine local and global models based on Markov Random Fields (MRF) which models the spatial relationships between points. Line segments between points are used as the basic descriptors of the face shape. The local models efficiently select the best candidate points, while the global model regularizes the result to ensure a plausible final shape.

Due to the fact that ASMs do not include texture information, ASM are robust to illumination variations. However, texture encode information about identity, skin color, wrinkles and furrows which might be important for expression recognition task and other domains of application. In addition, the ability to encode texture information makes the model able to generate texture, thus this enlarges the usability of the model in the synthesis field.

1.1.3 Constrained local appearance models

This category comports those approaches that constrain local models of appearance by global spatial models. Among these the Constrained Local Models, part-based model and the graph based model.

1.1.3.1 Constrained Local Models

A successor of ASM is the Constrained Local Models (CLM). Instead of sampling one dimensional profiles around each feature point, a square region is sampled forming local textures. "Feature detectors" are built based on these local patches for each landmark. [CC04] train three feature detectors: a normalized correlation detector by averaging over the training base and scaling such that the pixel values have zero mean and unit variance resulting in a fixed template, an orientation map detector where sobel edge filter is applied on the averaged template images and a cascaded boosted classifier. When searching in a new image, detectors are applied on specific search regions of each feature and response images are obtained. The shape parameters are optimized so as to maximize the sum of responses. Instead of using fixed templates during the search, [CC06b] updates the templates through the use of Nearest Neighbor (NN) selection approach. The NN scheme selects the most appropriate templates among the set of training templates. [CC06a] later extends this to build a joint model of shape and texture. Local patches are concatenated to form one vector and PCA is applied on shapes and textures to get shape and texture parameters. Another PCA is applied on the concatenation of these parameters to form appearance parameters. During the search, optimization of appearance parameters generates templates. Then response surfaces are computed by correlating to these templates. The search then proceeds by optimizing a function of the shape parameters until convergence takes place. [SLC11] propose an extension to fitting CLM through the use of Regularized Landmark Mean-Shift (RLMS).

CLM was reported to be efficient in tracking facial features with respect to global approaches that model the texture as a hole such as the Active Appearance Model (see section 1.3.1.4).

1.1.3.2 Part-based models

Part-based models also called pictorial models represent the face as a collection of parts of which are arranged in a deformable configuration. Each part's appearance is modeled separately, and the deformable configuration is represented by spring-like connections between pairs of parts. Most part-based models restrict these connections to a tree structure [FH05] which represents the spatial constraints between parts. Recently, such models proved to be very efficient in localizing facial landmarks. For instance, [KS11] uses such scheme. First parts are detected based on appearance based matching procedure. Once the parts are detected, each one of these is modeled using Histogram of Oriented Gradients (HOG). Separate linear regression models are then learnt on these parts to localize

landmarks. These regression models map each of the parts' appearances to the locations of the landmarks that exist in the corresponding part. Compared to global models (ASM and AAM) these models do not need any initialization however the method is dependent of the parts detection. In addition the number of parts and the landmarks assigned to each part were manually determined.

A very promising approach is that of [ZR12] which does simultaneous detection of the face, pose and shape. The authors proposed to model each landmark of the face as a separate part. To deal with the pose of the face, global tree mixtures of the landmark parts are used. Each facial pose represents a tree structure mixture of landmarks (parts). Parts are shared among different poses. The appearance of a part is modeled using HOG, and the shape is defined in terms of relative displacements between parts. The method shows robust results in the detection for large poses in unconstrained environments. However, concerning facial deformations, sometimes the model leads to unnatural deformations. This is because the tree structure that poses constraints on the global structure does not contain closed loops and thus the positions of feature with respect to each other are not modeled.

1.1.3.3 Face Graphs

Maurer and von der Malsburg [MVdM96] used graphs to track heads through wide angles and recognize faces. A graph for an individual face is generated as follows: a set of salient feature points are chosen on the face. Each point corresponds to a node of a full connected graph, and is labeled with the Gabor filters' responses (jets) applied to a window around the fiducial point. Arches are labeled by the distance between the correspondent feature points. The method was not used to track face deformations. [WFK97] also used the same kind of graphs for face recognition. They combine the face graphs to form a stack-like structure called the face bunch graph. Graphs were matched to new faces using Elastic Bunch Graph Matching.

[CBB02, GCJB03] propose an inexact graph matching problem formalization. The model of the face is represented by a graph, and the image to match the model to is represented as another graph. These graphs are built from regions and relationships between regions. Vertices correspond to different feature regions of which are assigned attribute vectors, and edges to relations between them. A global dissimilarity function is defined based on comparison of attributes of the two graphs, and accounting for the fact that several image regions can correspond to the same model region. This function is then minimized using several stochastic algorithms. The model had proved to efficient for following the motion of facial features. However, it failed when sudden changes occurred and for complex facial movements such as those of the mouth.

1.1.4 3D feature-based model analysis

To analyze the face deformation, some approaches tend to fit a 3D parametric model to the face. The problem is seen as an inverse problem where the parameters describing the current facial deformations in the images are extracted. Methods in this class proceed in two steps: 2D facial feature points extraction and tracking, followed by 3D parameters inference based on these points. An optimization process is needed to match the 3D points of the model to the 2D features. 2D facial feature tracking can be performed using any of the facial analysis methods described previously in the above sections. The advantage of such approaches over the above ones is that in addition to the shape output describing the facial features due to the 2D extraction, parameters encoding facial deformations are given. Often the used 3D models are models conceived for facial animation.

[TW90] first tracks the features of the face using snakes. To obtain precise alignment, the author enhanced the image by wearing makeup. Estimation of the muscle contractions of a physically-based muscle model (cf. section 1.2.3.2) is then done by interpretation of the state variables of the snakes in successive image frames. [RS08] first estimates the features positions using optical flow. The facial deformation parameters are then estimated using displacement-based Finite Element Models (FEM). [GL10] uses optical flow to track the feature points. Levenberg-Marquardt (LM) optimization algorithm is then used to fit the Candide 3D model (cf. section 1.3.2) to the face which results in the pose parameters. The latters are then refined by a template matching algorithm. Action parameters of the mouth and eyebrows are estimated by template matching using the same optimization. [HFR+11] infers the 3D parameters of Candide starting from the 2D landmarks found using a cascaded optimization algorithm of a 2D shape model. They optimize the model parameters by minimizing an image energy and an internal one. The first attracts the projected model to the edges of the face and the second imposes temporal and spatial motion smoothness constraints on the model similar to fitting snakes. [Lim10] uses AAM and Active Shape Models (ASM) to find the 2D features of the face. The 3D coordinates are projected on the 2D mesh and then an Euclidean distance between the projected coordinates and the 2D ones is minimized while tuning the Candide parameters to match the 2 meshes. [Bai10] labels manually the 2D points of the face. The author finds the Candide face pose by comparison to a set of synthetic images.

The disadvantage of these algorithms are that they are dependent of the features localization method. If the latter does not give accurate results, then the resulting parameters describing the facial deformations will be noisy. In addition, the need of a prior method to fit the model, complexify fitting 3D models to faces. We present in section 1.3 techniques that analyze the face through parameters without the necessity of a prior shape extraction method.

1.1.5 Discussion

ASM can be seen as a snake that has the ability to constrain its shape to a specific object. ASM and CLM methods belong to the so-called statistical learning family, since they incorporate a learning database in constructing their models. These two use texture information in their alignment procedure, but they lack a global texture model. Face Graphs and Part-based models are similar in the use of graphs to link nodes corresponding to salient points. These two classes relate to CLMs in that they constrain local models of appearance by global spatial models. All of the methods presented in the analysis section except for section 1.1.4 are not able to synthesize faces since they only model the global shape or local textures. In the following, we present a review on some facial expressions synthesis techniques used in the computer graphics community.

1.2 Synthesis models

Facial animation is a very active domain in computer graphics. The objective is to reproduce expressions on a face model in order to recreate emotions and animations corresponding to speech or sound.

Animating a face realistically requires tackling several aspects. First, a face model that models the face in its neutral state (no expression) together with the facial deformations should be implemented. Second, the temporal aspect of facial deformation also needs to be modeled. This second aspect concerns modeling the temporal trajectory of facial deformation, and how these deformations are related to each other over time. Among these two aspects we will concentrate on the first one. We revise face models and animation techniques that are used abundantly for animating faces in the synthesis field. These can be distinguished into three types: blend shapes, skeletal and parameter-based.

1.2.1 Blend shapes models

Figure 1.2: A virtual character's blend shapes. These key shapes are used to interpolate in between expressions.

The principle is to create several key topologies of the face where each topology represents an expression and then automatically blend these topologies by automatic inter-

polation in between them. Thus intermediate topologies or expressions are interpolated starting from extreme key expressions.

Typically, a linear interpolation function is used to predict the smooth motion between one topology and another. A simple example is the following. Given two key topologies, for example, neutral topology and wide-smile topology, an intermediate-smile topology can be calculated as: Intermediate Smile = (1-α)$Neutral$ + $\alpha WideSmile$, where α is the control parameter of interpolation. Generalizing to more than two dimensions, an in-between vertex is calculated as

$$v = \sum_{k=0}^{n-1} \alpha_k v_k, \alpha_k \in [0, 1] \tag{1.1}$$

where v_k is the vertex of the k^{th} blend shape topology, and α_k is the weight corresponding to it. Using α_k, one blend shape can be given more importance than the others. Figure 1.2 is an example of this animation technique. Even though linear interpolation methods [PHL+06] are efficient. However simple linear functions may not be able to accurately mimic the facial motion due to the complex curved topology of the face. That is why other interpolation functions such as cosine, bilinear functions and splines are used. Another extension of blend shape models is the hierarchical approach [JTDP05] that blends several models with differing weights on selected areas of face models.

The advantage of such technique is in the low computational cost due to the automatic interpolation in between frames. However, the disadvantage is that a large database of key topologies (expressions) is needed in order to be able to conceive a large variety of expressions. It is impossible to create an expression that does not exist in the key topologies database. In addition, due to the necessity of manually designing many key topologies, this technique might not be convenient for animating long sequences or real time applications that permit the interaction with the computer.

1.2.2 Skeletal-based models

Instead of blending different key topologies, another technique that is widely used in animation is Bones rig animation, also called skeletal animation. The principle is to rig a skeletal setup (hierarchical set of interconnected bones) that is bound to a 3D mesh. Like a real skeleton, each rig which is formed of joints and bones can be used to bend the character into a desired pose. Generally a rig is composed of both forward kinematics and inverse kinematics parts that may interact with each other.

Skinning is the process of associating each bone with some portion of the character's visual presentation. Each bone is associated with a group of vertices. Some of which can be associated with multiple bones such that they are influenced by the actions of these bones and not only by the action of one bone (cf. figure 1.3). Each vertex have a weight associated for each bone. To compute the final position of the vertex, each bone transformation is applied to the vertex position, scaled by its corresponding weight. The

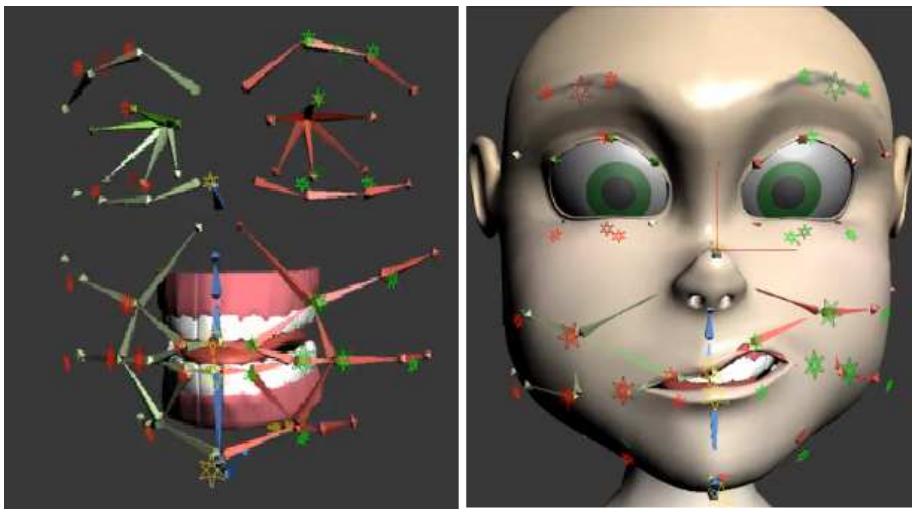


Figure 1.3: An example of skeletal animation of the face

eyes can be outside of the skeleton hierarchy and skinned to the eye joints. Figure 1.4 shows an example of eyeballs rigging using an interactive animation software "Blender". After setting the rigs of the eyeballs, they are rigged by rotating the rigs around some pivot.

Rigging a character is done through the use of an interactive tool. Nevertheless, recently automatic rigging is employed. For instance, [BP07] automatically adapts a general skeleton into a character and then animates it using skeletal motion data.

The advantages of skeletal animation over blend shape animation is that it is less labor-intensive. Actually in blend shape animation, every vertex should be manually manipulated to produce animation, which limits the number of blend shapes. Whereas in skeletal

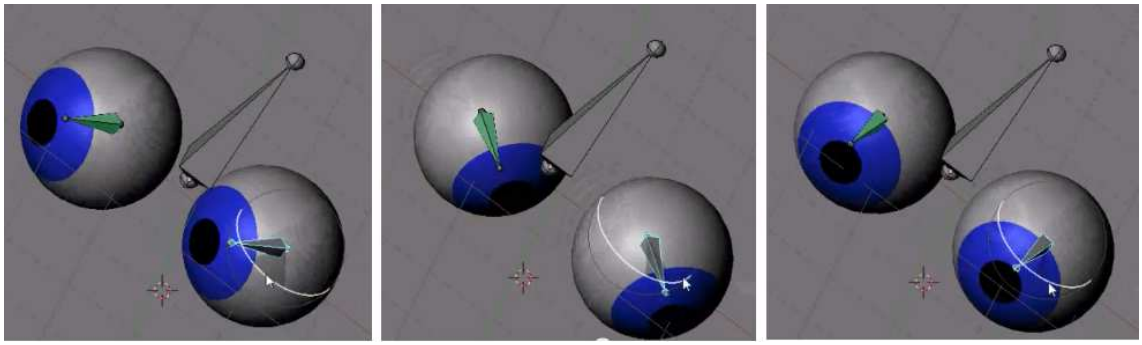


Figure 1.4: An example of rigging the eyeball using the "blender" animation software

animation, the movement of vertices is simply done by moving the skeleton. On the other hand, skeletal animation is convenient to animating body movements. However regarding skin and facial expressions it might produce unrealistic movements, since conforming these to bones needed for skeletal animation is not easy.

1.2.3 Parameter-based models

Another type of approaches to animating characters, consists of using a system of parameters to manipulate the facial topology. Using a system of parameters permits to make animations that are independent of the used topology. It also becomes possible to design expressions or animations interactively by producing these parameters using different approaches based on text, audio or video. Actually, the blend shape method previously described does not deal directly with the deformation of the face. It only deals with the combination of different deformations which are defined manually. Instead, parameter-based methods can be used to create different deformations corresponding to the key shapes, and then blend shapes technique can be used to create different animations.

Different approaches for modeling facial deformations through parameters can be found in the computer graphics community. Some deal with the simulation of the visual effect of the muscles (pseudo-muscles approaches) using geometric approaches. Others deal with the physical simulations of the muscle actions (physically-based muscles).

1.2.3.1 Pseudo-muscles geometric models

From a biomechanical point of view, the activation of facial muscles causes the deformation of the skin, which results in the facial expressions that we see. A variety of approaches have tried to simulate the visual effect of the muscles on the skin surface, without dealing with the underlying structure, through the use of purely geometric techniques. Such methods are called pseudo-muscles technique.

1.2.3.1.1 Direct Parameterizations

Models providing a set of parameters that directly manipulate the vertices of the face geometry are referred to as "Direct parameterizations". [Par74] was the first to propose a parametric model of facial deformations. His model parametrized the face using a set of approximately ten parameters to control the morphology of the face and around twenty parameters to deal with expressions. Later on, approaches tending to parameterize facial deformation increased abundantly. [Par82] used different procedures to animate the different parts of the face mesh. Concerning regions of the face that change shape, interpolation between predefined extreme positions was used. A parameter is specified to control the interpolation. Scaling was used to control some facial characteristics such as the mouth width. The mouth was opened using rotation around a pivot. Translation controlled lip corner and upper lip opening. Eyeballs were assigned a different mesh from that of the

face. Their animation was done by a procedural construction of which polygons descriptors of the mesh are generated according to the eyeballs parameters. Direct parameterizations of the face are efficient but simple and are not able to model complex and subtle facial deformations. The resulting animation is thus not that realistic.

1.2.3.1.2 Elementary deformations based models

Elementary deformations based models are those that are based on well-defined elementary movements of some predefined points of the facial mesh. Such methods need a formal description of the facial movements. They provide a high level of abstraction of facial motions.

The Facial Action Coding System (FACS) invented by the psychologues Ekman and Freisen [EF77] is a system that formally describes the elementary facial movements through the analysis of the facial anatomy. Their work is based on the observation of the effect of each muscle on the facial appearance: a facial expression is the combination of a set of facial actions caused by the simulation of one or several facial muscles. Their study led to the decomposition of the visible movements of the face in terms of 46 Action Units (AUs) that correspond to facial actions which describe the elementary muscular movements (for example AU1 corresponds to raising the inner eyebrow). Each facial expression can thus be represented as a combination of Action units (AUs): A sad expression is the combination of AU1 (Inner Brow Raiser), AU4 (Brow Raiser), AU15 (Lip Corner Depressor, and AU23 (Lip Tightener)).

Though FACS was not originally created for the use of facial animation, it was later adopted by computer graphics animators for this goal. Actually, it does not provide quantitative definition of AUs. Thus, animators had to generate these using their own ways. For instance, [TH99] uses FFD Bernstein polynomials to simulate AUs.

Candide model [Ryd87] is one implementation of the AUs presented in the FACS. It is a generic face model with Action Unit Vectors (AUVs) to implement the AUs and animate the face and Shape Unit Vectors (SUVs) to model its static properties. This model is detailed in the Analysis-by-Synthesis class of models in a dedicated section (1.3.2).

The **MPEG4** video coding standard also presents a face model for animation based on elementary deformations of some Facial Feature Points (FFPs). These FFPs are associated with a set of Facial Animation Parameters (FAPs) that describe the facial actions. Measures on these FFPs are made to form units of measure (Facial Animation Parameter Units (FAPU)).

FAPU permit to define elementary facial movements having a natural aspect and they serve to scale FAPs for any face model. They are defined as fractions of distances between key feature points. Actually, it is difficult to define elementary movements of muscles in an absolute manner: the absolute displacement of muscles changes from one person to another, but their relative displacement to certain pertinent measures is constant. This permits to animate the faces in a realistic manner and can permit to give human

expressions to non human avatars.

As examples of FAPU, we can cite the mouth width, the separation distance between the mouth and the nose, the separation distance between the eyes and the nose, etc. For example the stretching of the left corner of the lips (Facial Animation Parameter 6 $stretch - l - cornerlip$) is defined as the displacement towards the right of the lips corner by a displacement that is equal to the mouth length. Thus, the FAPU are measures that permit to describe the elementary movements and thus the animations.

However, the Facial Animation Parameters (FAP) of MPEG-4 do not represent directly realistic movements of the face, contrary to FACS. FACS describe a group of muscular movements, while MPEG-4 describe a group of visual movements that are not necessarily realistic. For example, the Action Unit (AU) 26 of FACS (jaw Drop) describe the movements of lowering the jaw ; this lowering is accompanied by a lowering of the lower lip. The lowering of the jaw of MPEG-4 (FAP 3 $open - jaw$) does not describe the lowering of the lower lip: the description is thus not realistic from a muscular point of view. And so, we can consider that the FAPs of MPEG-4 are low level descriptions of the AUs of FACS.

Later, J. Ahlberg [Ahl01] made the Candide model compatible with the animation model of MPEG4 through the third version of Candide.

Abstract Muscle Actions (AMA) proposed by [MTPT88] are control procedures also conceived to simulate the muscular actions based on empirical geometric movements. Just like the AUVs, each AMA corresponds to an action of a simple muscle or a group of muscles, and works on a specific region of the face. A procedure is associated with a set of parameters that are responsible for simulating the muscular action. For example, the $COMPRESSED_{LIP}$ procedure simulates the action of the orbicularis oris muscle (a kiss) by employing parameters that control the inside and outside compression amplitude of the corners of the mouth, and the advancement of the lip vertices in the z-direction. Compared to AUs, these facial actions are not independent from each other, and thus they guarantee more realistic animations.

Minimal Perception Action system of [Kal93] is a similar system to the AMA. It contains a set of normalized parameters in the range of $[-1, 1]$. Some of these parameters are responsible for elementary muscular movements, others correspond to non muscular actions such as the rigid movements of the head. The activation of each MPA simulate a specific visual effect on the face. Rational Free Form Deformation [KMTT92] is used with this system to animate the face. Free-form deformation (FFD) [SS86] is a geometric technique that is based on enclosing the face within a $3D$ lattice of control points in the form of parallelepiped. The face is deformed within the lattice as the control points are deformed. The animation consists of three steps. First, a $3D$ lattice around the face is created and each point of the face is assigned local coordinates. This plays the role of a local coordinate system. Second, a grill of control points is imposed on the lattice. Finally the face is deformed by moving the control points. The deformation of a point of the face is thus a function of these control points.

Figure 1.5: Rational Free Form Deformation for simulating muscle movements to animate faces [KMTT92]

In the Rational B-splines of [KMTT92], each of the control points is assigned a weight that defines the attraction of the point on the surface. The authors divide the face into different regions based on anatomical considerations on which a muscle action is desired. Muscle actions are simulated by displacing the corresponding control point and by changing their weights.

Example-based deformation All of the above parameterizations are done manually and are based on human experience and knowledge. A more confident approach is to try to drive expression parameters starting from real data. In such approaches, a capture of an actor performing facial movements takes place. The actor either wears markers on his face, or manual annotation of the sequence is performed afterwards. Statistical analysis techniques are performed to find basis of expression deformations.

For instance, [Sto10] uses Principle Component Analysis (PCA) in the context of an Active Appearance Model (AAM) to derive a parametric representation of facial deformations. PCA is a statistical tool that decomposes facial expressions into a set of deformation basis. AAM uses PCA to model deformations in shape and in texture. Geometric and photometric deformations are thus coded by the appearance parameters of AAM. The authors map the appearance space into an expression manifold in the form of a disc. Dominant directions in the appearance space are automatically identified and then associated to the manifold. PCA succeeds in parameterizing facial deformation, however the resulting parameters are not interpretable. It is not straightforward to interpret which parameter corresponds to what deformation. As an alternative, [SL09] uses Independent Component Analysis to decompose the expression into a set of independent deformation modes where each deformation mode corresponds to a specific facial movement.

[FKY08] trains a predictor that predicts surface deformations together with bone deformations in a skeletal animation starting from a set of examples. Given a set of training meshes with different deformations, the authors semi-automatically choose a set of representative sparse key points on the mesh using a PCA combined with Varimax rotation

scheme [MA07]. The corresponding bone deformations are then automatically computed and a predictor on the pairs of the points and the corresponding bone deformations is trained. At run time, new deformations are produced using those learned from the training examples.

1.2.3.2 Physically-based Muscle models

Some facial animation models try to approximate the skin deformations by simulating muscle contractions through the use of physical models. These models use dynamic equations to model the muscles movements, thus the deformation of the face is determined by solving these equations. [PB81] modeled the behavior of the skin according to muscles actions through the use of a mass-spring network model. In such model, the face is represented as point masses connected by springs. Skin deformation is simulated by forcing these points into elastic spring mesh. Another physical muscle methods used *muscle vectors* [Wat87]. In such methods the facial mesh is deformed according to directional muscle vectors that move in $2D$ and $3D$ directions with a certain magnitude.

To simulate volumetric effects of the face, [TW90] proposed a *three-layered spring* mesh for modeling the detailed anatomical structure and dynamics of the human face. The three layers correspond to the skin, fatty tissue and muscle which are tied to bones. Another type of physically based models are those that build an elastic thin shell continuum mechanical Finite Element Model (FEM) [EBDP96].

Compared to pseudo-muscle parameterizations, physically-based muscle models are more powerful in producing realistic facial expressions since they model the face in detail. However, the computational cost of expression synthesis is important compared to other approaches.

1.2.4 Discussion

Among the face modeling and animation techniques of the computer graphics community, we find that there is an interesting aspect. It concerns the parametrization of the facial movements in a way that each facial action is identifiable and assigned a well-defined parameter. Another aspect is the way the eyes are modeled. As we have seen in the skeletal animation section (1.2.2), eyeballs are treated as separate objects from the skin mesh. In chapter 3, we make use of this approach in the research work presented in this thesis.

1.3 Analysis-by-Synthesis models

w

Analysis by synthesis approaches rely on the synthesis of an image using a model by varying a set of parameters. The positions of the different features of the face are computed by comparing the synthesized image to the real one. The optimal parameters of the model are those that best minimize the difference between the two. The particularity of such

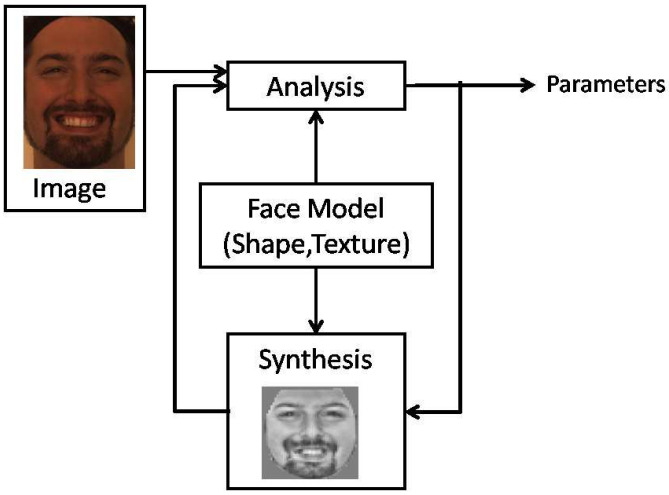


Figure 1.6: Analysis-by-synthesis loop

approaches is that they serve for both analysis and synthesis purposes. They constitute a kind of loop of which synthesis is established by the analysis and vise versa (cf. figure 1.6). In the literature, we distinguish between the statistical models and the manual models.

1.3.1 Statistical models

Statistical models are those that are built starting from real data and based on statistical tools. Among these models, we have eigenfaces, active blobs, morphable models, and active appearance models.

1.3.1.1 EigenFaces

[TP91] proposed the eigenfaces for the purpose of face recognition. Principle Component Analysis is performed on a set of learning images having the same size. Variations in pixel intensities are thus learned and coded in terms of a set of basis functions (the "eigenfaces"). This approach has made a fingerprint in the domain of face recognition. However, since eigenfaces are made only on texture information, they are not very robust to shape variations. Also, they are not known to be robust to facial pose changes and expression variations.

1.3.1.2 Active Blobs

In active blobs [SI98], the model is constructed from one example starting from an interactive user interface. The user circles the object of interest in the image. Delaunay triangulation is applied to the resulting shape. Texture under the mesh is mapped to form a

texture model. Two ways can be used to model Blob deformation. Either it is described in terms of orthogonal modal displacements (Finite element modes). In this case, the system's shape is modeled as an elastic material. Or statistical analysis through PCA is performed on the previously collected data samples of displacement of each node of the shape. To model the appearance variations due to lighting, an illumination basis based on Taylor series approximation is used. Tracking a deformable object using Active blobs constitutes minimization of some objective function which includes a geometric deformation energy term and an image difference energy term. The first term is a sort of a regularization term, it measures how much energy it takes for a blob to deform into its current shape. The second term measures the similarity of the mapped texture to the initial image of the tracked object.

Active blobs are able to follow non-rigid motions of the face such as the motion of the eyebrows [LCIS98]. However, they are not effective for tracking head pose. They were not tested to track complex non-rigid motions of the face such as the motion of the mouth.

1.3.1.3 3D Morphable models

A morphable model is a 3D morphing function that is based on a linear combination of a large number of 3D face scans [BV99]. This model was conceived for both synthesis and analysis purposes. The idea is to represent a face as a function of a set of basis functions and to constrain the faces by a set of predefined faces. This is similar to the blend shapes facial animation technique using a data driven representation where statistical tools are used to project the shapes on a higher dimensional space.

Starting from laser scans which provide dense color and shape representation of faces (in the order of thousands of vertices), a statistical model of shape and texture is constructed after performing a dense correspondence of the scans to each other. Shapes and textures are thus defined to be the sum of the mean shape or texture plus a linear combination of a set of linear basis. The resulting model is able to reconstruct any face by varying the model's parameters. Dense representation of initial scans gives abilities to reconstruct face details.

The original model of Vetter does not totally remove pose effects before performing correspondence (only coarse removal of these is done). [PS09] contributes to this by aligning the dense shape using Procrustes analysis which aligns the faces to the mean shape that is iteratively computed during the alignment process. This contribution results in a more efficient and accurate Morphable Model.

Even though 3DMMs are very robust, however, the need to process a set of very dense shapes makes them computationally expensive. This hinders real time applications. In addition, the need for laser scanners to obtain 3D shapes and textures is indeed a disadvantage, since this requires additional hardware.

1.3.1.4 Active Appearance Models (AAM)

AAM is similar to active blobs and to morphable models. The difference with the first is that in the active blob, the model is generated using 1 example whereas in AAM multiple examples are used. Compared to 3DMM, the latter uses shapes that have much higher number of vertices. MM attributes a higher number of parameters including camera-specific, light-specific, color-specific and image-specific parameters which makes the MM fitting a very long process. This model can be seen as a combination between the ASM (section 1.1.2.2) and the eigenfaces (section 1.3.1.1). After modeling the shape variations similar to that in ASM to get the shape parameters and modeling texture variations as in eigenfaces and obtaining the texture parameters, AAM fuses the resulting parameters in a final step. This final joint modeling of the shape and texture parameters is specific to AAM making the model capable of encoding the appearance and the shape of the face in the same vector of parameters. In this thesis, we choose to use AAM. As a matter of fact, despite of certain limitations, such model has proved to be simple and efficient. We detail AAM formulation together with the limitations that this model presents in the next chapter. We thus propose solutions in this thesis in an attempt to tackle these limitations.

1.3.2 Manual Models

Manual models are those that are built based on human knowledge and observation. Among these, we cite the Candide model which is built in conformation with the AUs of FACS.

We have already mentioned Candide model in the face synthesis section (1.2) since it is a model that directly implements the Action Units described in FACS for the purpose of face animation. However, this model was widely used in facial analysis to align it and find the parameters describing the face. Particularly, a number of researches have integrated Candide in an analysis-by-synthesis paradigm to achieve this purpose. Candide offers a formal description of facial movements based on AUs of FACS. Thus, each facial movement has a Candide parameter corresponding to it.

Figure 1.7 shows the Candide model with facial actions corresponding to some non rigid facial deformations.

Shape modeling – Candide offers a formal parametrization of the facial shape. It is constituted of a standard 3D shape of which the vertices are stored and defined in a local coordinates system. Shape deformations are coded in a well-defined unit vectors called the Shape Units Vectors (SUVs) and Action Unit Vectors (AUVs). The formers are responsible for static person specific shape characteristics such as the eye separation. The latters are responsible for the facial features elementary movements such as opening the mouth. They are implementations of the Action Units (AUs) derived from the study of Ekman and Friesen [EF77] on the physiology of facial expressions and defined in the FACS (Facial Action Coding System).

An AUV and an SUV are represented as a set of elementary vertex displacements in

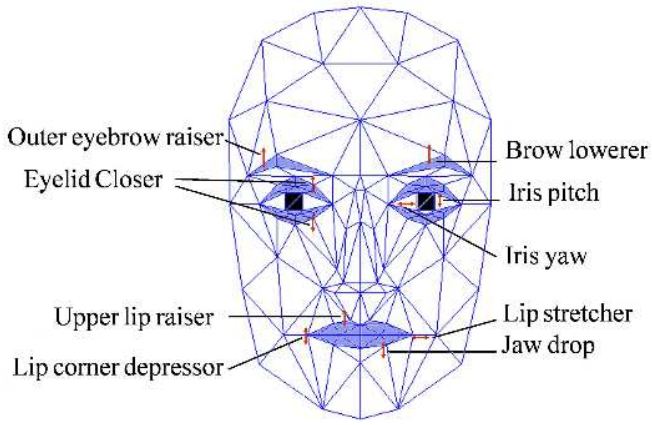


Figure 1.7: Candide model with 9 Facial actions (from [Oro07])

a local coordinates. They were defined manually and are measured in pixels. A simple example is AUV 10, the "Upper lid raiser", which is an implementation of the AU 5 of FACS, shown in table 1.1.

Vertex	X	Y	Z
21	0	0.03	-0.01
54	0	0.03	-0.01
97	0	0.015	-0.007
98	0	0.015	-0.007
105	0	0.015	-0.007
106	0	0.015	-0.007

Table 1.1: AUV10 of Candide model: implementation of AU5

Using these SUVs and AUVs, any expressive face can be decomposed, and any neutral face can be animated using the following equation.

$$s = \bar{s} + S\sigma + A\alpha \tag{1.2}$$

where $\bar{s}$ is the standard shape of the Candide model. $S\sigma$ represents the static part of the model. S is the shape unit matrix. The columns of S contain the SUVs. The linear combination of the SUV weighted by σ represents a personalized neutral face. σ is a column vector that contains the shape parameters (SP), they represent the amount of variation added to the neutral Candide to form the shape of another person. Figure 1.8 shows some examples of varying shape parameters.

On the other hand, $A\alpha$ represents the dynamic part of the model. In other words, the deformation done by the face. A is the animation unit matrix with its columns

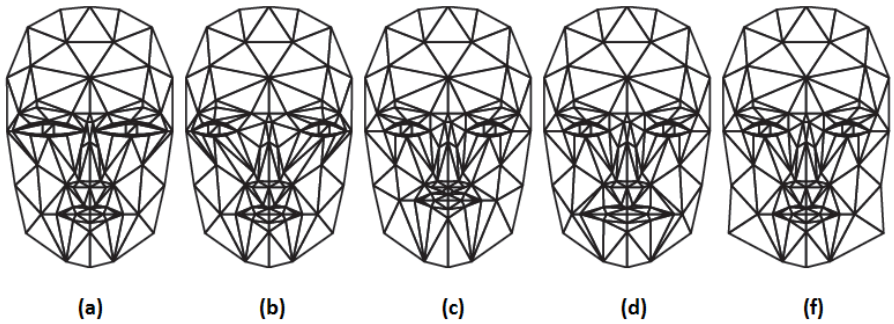


Figure 1.8: The effect of changing shape parameters; (a) original CANDIDE-3 Model. (b) Head height. (c) Eyebrows vertical position (d) Eyes vertical position (e) Mouth vertical position (f) Eyes width.

containing the AUVs. α is a column vector that contains the animation parameters (AP) which control the facial deformation. These animation parameters are in the range $[0, 1]$ where the 0 value corresponds to the neutral position (no deformation), and the 1 value corresponds to the maximum deformation. Figure 1.9 shows some examples of varying action parameters.

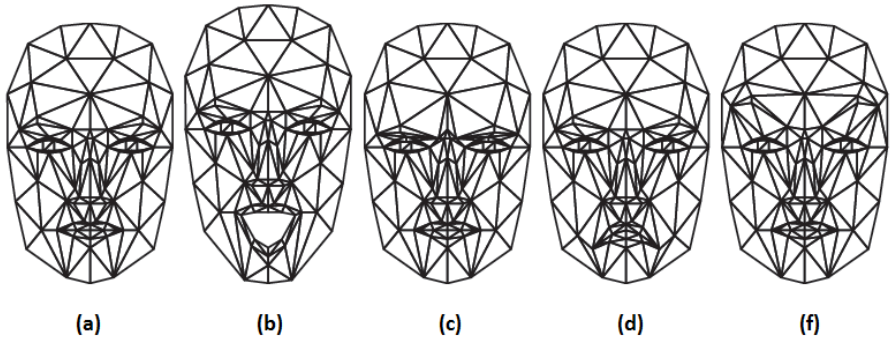


Figure 1.9: The effect of changing 4 action parameters; (a) Neutral expression (b) Jaw drop. (c) Brow lowerer (d) Lip corner depressor (e) Outerbrow raiser.

Candide model can be scaled, translated and rotated as well using the $3D$ pose vector responsible for the global head motion.

Texture modeling – Since Candide was first introduced as a model for animation and not for feature extraction, it does not come with a texture model. As a matter of fact, it only models the face's shape excluding its appearance. In some methods, to fit the Candide model to images, a texture model or simply a texture is needed. [Ahl02, YZ07] model texture by eigenfaces (same as they are modeled in AAM). First, Candide model is fitted to a number of images manually. Texture under the shape are then mapped on the

model and then normalization to a standard shape is done. PCA is then performed on the resulting texture to find the texture modes of variation. [MMKC08] compute a reference texture by finding the mean texture of some images where the model was fitted manually. Supposing that the subject is expression-less at the first frame, the reference texture is then replaced by the subject's texture which is obtained after adapting the model to the expression-less frame of the subject. [DD04, Hér07] model the texture as a multivariate Gaussian distribution after extracting the shape normalized texture.

Candide fitting – Fitting Candide using an analysis-by-synthesis loop resembles fitting AAM. As a matter of fact, the same rubrics are needed, that is: a reference shape, a reference texture or a texture model, an energy term, and an optimization algorithm.

[Ahl02] uses the Active appearance training and search process to fit the Candide model to images. Pose and action parameters of Candide are trained whereas the shape parameters are not (manually tuned on the first frame). [YZ07] also uses the same methodology as [Ahl02]. The difference is that he uses Canonical Correlation Analysis instead of linear regression to compute the experiment matrices. For Fitting unknown subjects, the reference texture is computed from training examples. [DD06] uses a particle filter based method to estimate the head pose and facial actions using Candide. They propose three methods. The first one models the texture statistically by applying PCA on training images. The second one models the texture using a multivariate Gaussian. The third one uses a combined exhaustive and directed search to optimize the model parameters of the second proposition.

[MR11] utilize a "displacement expert" approach to fit the Candide model. Facial component feature bands that reflect the location of the facial features are extracted from the image, in addition to the raw image data that are used to learn the "displacement experts". The difference with the above fitting approaches is that no non linear optimization is needed for fitting and thus no linear assumption is required.

In [MMKC08], fitting is done by minimizing the cross-correlation between the reference texture and the projection of the Candide model on the image and its derivatives using a multi-scale gradient descent algorithm. The drawback of this method is that it is time consuming due to the optimization process and that the texture model is not very sophisticated. [WVS+06] deals with the fitting of Candide as a registration process by employing iterative closest point (ICP) used to align Candide to scans. The Action parameters of Candide are tuned using exhaustive search whereas the pose is tuned using Procrustes analysis. [WULO06] adapts the Candide model to face scans using interpolation based on radial basis functions. Manual landmarks are initially placed on the scan. The pose parameters of Candide are estimated using a differential evolution optimization scheme.

1.3.3 Discussion

The difference between the statistical approaches and Candide is that the statistical models result in a set of parameters that are learned from real data, while Candide is a manual model that parametrizes that facial movements based on observation. With

statistical models, real deformations can be learned, and thus more realistic deformations. This is why statistical models are more preferable to manual ones.

Between AAM and 3DMMs, the former is simpler to realize since it does need any expensive requirements such as scanners. In addition, due to its speed and simplicity, AAM is more convenient to real time implementations. This makes it possible to use AAM in real scenarios such as a user in front of his laptop and using his ordinary webcam.

As a conclusion, due to the advantages of AAM over the other models in terms of efficiency and speed, we place our research in the field of statistical modeling using Active Appearance Models.

1.4 Gaze and blink detection: a state of the art

In this section, we review literature concerning gaze and blink detection due to their importance for human computer non-verbal communication mentioned in the introduction of this thesis.

1.4.1 Gaze tracking

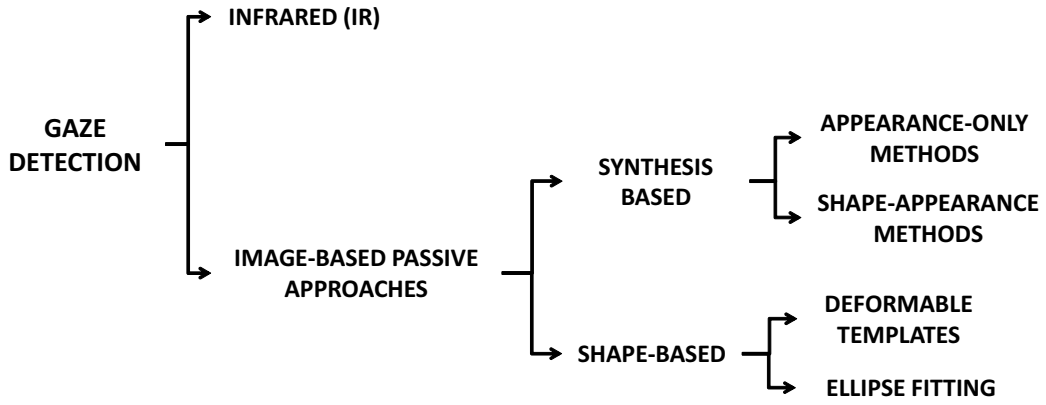


Figure 1.10: Flow chart of the state-of-the-art classification of gaze tracking

Research in this area is very active and many methods exist in the literature. These methods may use Infrared light or traditional image based passive approaches [ZJ05b]. A detailed state-of-the-art on eye and gaze tracking techniques can be found in [HJ10]. Figure 1.10 depicts our classification of the state of art.

It is very important to differentiate between three problematics:

1. Eye localization that is localizing the eye region. This can be used for further processing of the eye region.

2. Eye tracking that is locating the iris position in the eye
3. Gaze tracking that is using the information from the eye localization or the eye tracking to determine the person's gaze

In the following, we concentrate on methods that aim to find the iris location in the eye. Then we review some methods that integrate the head pose in the amelioration of this infromation.

1.4.1.1 IR methods

Many eye tracking and gaze estimation methods use active infrared illumination [HAS11] for estimating the position of the iris or pupil. If the light source was placed near the optical axis of the camera, the retina reflects most of the light to the camera and thus a very bright pupil (brighter than most of the objects in the image) appears. In contrary, if the light source is placed far from the optical axis, the pupil appears very dark in the image. These characteristics are often used to track pupils in the image since they facilitate this task. In addition, the glint which is a corneal reflection of the emitted light source and which appears very close to the pupil is detected. For instance, many methods rely on the image difference method to detect the pupil where the difference between the dark pupil image and the bright pupil image resulting from switching between on and off-axis light sources is computed [Ebi04, PWL05, ZJ04]. Some authors integrate more than one IR light source in their systems [PCG+03, Ebi04, ZJ04]. The goal of using more than one IR source is to be sure that a glint is always present in the image.

Although simple and efficient, IR methods are strongly dependent of the brightness of the pupils which is influenced by many factors such as eye closure and occlusion, external illumination and the distance of the user from the camera. In addition, the requirement of IR sources for the corresponding algorithms to work is itself a constraint and such methods are limited to indoor use. Nowadays, the challenging issue is the iris detection in visible light since it is more adequate for the outdoor use and is more consistent with natural conditions.

1.4.1.2 Image based passive approaches

These methods use the eyes intensity or shape or their combination to detect the gaze. They can be classified into two categories: Shape-based and Synthesis-based.

1.4.1.2.1 Shape-based methods

Shape-based methods contain those based on deformable shape templates of the eye and ellipse fitting methods.

Deformable templates – Methods based on template matching use a template of the eye to find the location of the iris and eyelid points. The image is scanned to find candidates of the eye and then candidates are filtered using some similarity measure.

[CT07] and [YHC92b] use a deformable template of the eye composed of two parabolas for the eye shape and a circle for the iris. For the matching process, they use four energy fields composed of intensity edges, valleys, peaks and gray levels. [CKLO09] uses the elliptical separability filter to find iris candidates in the image where an eye template with elliptical eyelids and circular iris shapes is looked for in the facial image. The test of this method is done only on people in frontal view and looking in front of them so its efficiency in detecting different iris positions is not known. A similar approach is that of [KR03]. In this approach the similarity cost is calculated using Hough transform, separability filter and template matching. [XWLZ09] uses AAM to obtain a rough localization of the eyes. For the iris center, the author uses a circular shape of the iris. His approach is based on calculating the integral of the pixel intensities inside the circular area of the iris. The iris center and radius correspond to the partial differential maximum value of this integral.

The template based methods are generally effective but their disadvantage is that they are computationally expensive. They also necessitate the definition of an adequate set of initial parameters for the template. The template should be initialized close to the eye in order to give good results because the energy minimization process only finds local minimum. They usually record a failure with big head poses.

Ellipse fitting – Many iris detection or tracking methods model the iris and the pupil as ellipses or circles. The best fit is obtained by varying the model's parameters. [KK07] searches in the eye region to fit an ellipsoid to the iris. The center of the resulting ellipsoid is the center of the pupil. [IBMK04] computes an initial estimation of the iris position by template matching and refines this location by an edge-based ellipse fitting. [RWDB08] applies a Starburst algorithm (rays are shot from the center of the eyeball found using the $3D$ iris model) to iris and pupil segmentation which includes finding the best ellipse fit on feature points detected on these two by calculating the gradient along rays issued from an initial rough estimate of the pupil center. The points of the rays with the largest gradient peaks are points belonging to the iris and pupil. [HP03] uses the EM active contour algorithm to detect the iris position in the image frame. The iris is modeled as an ellipse. Such methods succeed at finding the location of the iris or the pupil in the eye on the condition that high resolution images are provided. Their drawback is in their incapability of coping with the different states of closure of the eyelids.

Other methods are edge detection based. [Khi10] proposed an iris tracker based on the Hough Transform. Their approach needs constant illumination and little to no head translation or rotation; as such, they do not tackle the problem of head pose. [VG08] used isophotes based approach. However, their method fails with closed eyes, very bright eyes, strong highlights on the glasses, eyes with variable lighting, and on large head poses. In their approach, the eye region is assumed frontal.

1.4.1.3 Synthesis based

Synthesis based approaches are those that rely on the synthesis of an image using a model and compute the position of the iris by comparing the synthesized image to the

real one. The optimal parameters of the model are those that best minimize the difference between the two.

[RCY+11] analyses the iris center movements by employing a $3D$ eyeball/iris model. His model contains the position of the eyeball, the iris radius and size. The eye image data is projected to the model and pixel error is minimized between it and the rendered rotated eyeball that gives different iris positions. Iris contour extraction is done using a Starburst algorithm. [YUYA08] also applies a similar head-eye model. It differs from that of [RCY+11] in the fact that the $3D$ model is projected on the image plane and not the contrary. The iris is modeled as a circle moving on the eyeball. The $3D$ eyeball model of [WKW+07] consists of the eyeballs, iris contours and eyelids. Particle filter is used to track the iris contours. [MKXC06] employs a very detailed eye model that describes both the appearance and the motion of the eye. The author assigns for every region of the eye a specific model. As for the motions of the iris and the eyelids, time dependent parameters are associated. The final eye model is a fusion of all of these models. Although accurate, this model is very complex.

As for the methods that use active appearance models for iris localization, a training phase of the model using a set of annotated images that include subjects with different gaze directions is required. [HNH+02] combines a mean-shift color tracker with a hierarchical AAM to track the eye corners and the iris positions. [BIC08] uses component-based AAM in order to model the different iris positions. The author uses a total of 80 landmarks in order to describe the eye region. [Iva07] deals with the occlusion of the iris by the eyelids by annotating the iris as a perfect ellipse where when occlusion exists, the landmarks points will lie on the eyelids. All of the methods for iris position detection that are based on AAM rely on the common fact that faces with different gaze directions should be included in the training set. In addition, if the model used is $2D$, different head poses should also be included in order to be able to find the iris when the face is not in frontal view. The more images are included in the learning base, the more parameters are necessary to describe the appearance of one face.

We propose a face model that deals with the movements of the eye region and is able to detect gaze. Compared to the above state of art, the proposed system works with low resolution images, it does not constrain the user with special requirements (IR illumination, hardware equipment...) and it makes use of the appearance and shape of the eye while avoiding explicit manual design of the model through the use of AAMs. With respect to classical AAM, it has the advantage of restricting the learning database of AAM to people in frontal view and looking in front of them where there is no need to learn on people with different gaze direction.

1.4.1.4 Head pose in gaze detection

All of the above mentioned methods neglect face orientation in the detection of the iris location. However, we can find in the literature methods that use iris location results to detect pose ([QX02]) or vise versa ([SY97]).

In [VSG12], head pose is used to enhance eye center detection. The eye regions are normalized by the transformation matrix obtained from head pose. Actually the idea is to pose-normalize the eye regions to enhance the eye center localization. [QX02] use the pupil detection results (based on IR illumination) to estimate the pose. They exploit the correlations between these two and build a pupil feature space (PFS) which is constructed by the characteristics of the pupils (pupils' sizes, intensity...) and their inter-distance. Based on the correlations between the pose and the pupils' characteristics, head pose is calculated by projecting pupil properties in the FPS. [RCY+11] also use head pose to find the position of the eyeball of which they use to detect the iris locations. Yet, very few integrate head pose to improve the eye localization except [VSG12] that normalizes the eye regions by the pose to improve eye center localization. [HSL11] perform gaze detection with respect to the head pose instead of the camera using blob detection for iris tracking.

In this work, we propose to use the information of the pose for the amelioration of iris localization through a multi-objective framework. We apply an iris localization algorithm simultaneously on both eyes and sum the resulting errors while multiplying each by a weighting factor that is a function of the head pose (more details are presented in chapter 3).

The following section is dedicated to literature about blink detection.

1.4.2 Blink detection

Blinking is a physiological necessity for humans. It is a periodic involuntary or voluntary action that we do during our daily lives. In addition, the movement of the iris is correlated with that of the eyelids, especially if the person is looking downwards or upwards or if he is scanning a scene with his eyes. Automatic blink detection is useful for HCI applications such as driver drowsiness surveillance in cars or computer control through blinking instead of the traditional interaction techniques (mouse or keyboard) [GBL+03]. Another application is the eyes surveillance of computer users in workplace so that they would be alerted if they do not blink their eye sufficiently. This helps avoiding chronic dry eyes which may lead to eventual sight loss [DB08]. Thus, tracking eyelids locations is not enough. A parameter that encodes the eyelids state (open, closed or intermediate state) can be very effective. This section reviews some of the blink detection and eyelids tracking methods of the state of the art. We classify these methods into: feature-based, motion-based, state-based and parameter-based (Figure 1.11).

1.4.2.1 Feature based methods

Many blink detection methods rely on the detection of the iris. These methods assume that if the iris is not found in the image, then it is occluded by the eyelids signifying the occurrence of a blink. For instance, [SNSM12] detects eye blink by searching for the center of the pupil. This is done by image smoothing followed by edge detection and morphological operations. If there was no detected pupil in the image, then the eye is

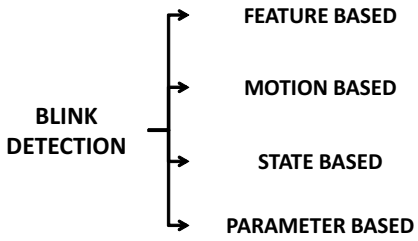


Figure 1.11: Flow chart of the state-of-the-art classification of blink detection

considered to be closed. A blink is then detected if there is 2 consecutive open and closed eyes. This method needs very high resolution images of the eye. In addition it is not capable of calculating intermediate eyelid states. [TKC00] also detects blinks based on the presence of the iris. The iris is detected based on intensity and edge information. If an iris is detected then the eye is considered to be open and an open eye template is used to recover the eye parameters. Otherwise, the eye is considered to be closed and a closed eye template (a simple line template connecting eye corners) is used. This method does not deal with intermediate eye states nor with head rotations.

On the other hand, other methods use the locations of eye features to detect the blink. For example, [SRD02] compute normal flow on the edge pixels of the head, iris and eyelids motion to identify the motions of these and drive corresponding models. Based on the flow vectors, the head is modeled separately and its corresponding motion is filtered from those of the eye motions. The positions of the iris and eyelids are then tracked based on models deduced from the motion flow. Blinking is identified based on the distance between the apex of the upper eyelid and the center of the iris.

The disadvantage of these methods is that they are dependent of the localization of the iris which is not an easy task especially in the presence of head rotations and when the person does not have a frontal gaze.

1.4.2.2 State-based methods

State-based methods are meant to detect the state of the eye: closed versus open. [PSWL07] formulate blink detection as inference in an undirected conditional Random Field framework. They incorporate a discriminative measure of eye states derived from the adaptive boosting. [AS09] detects the eye region based on a Haar-like features cascade classifier. Blink is then detected using an iterative thresholding scheme. The method iterates on the value of the threshold until reaching a value that keeps at least one black pixel after blurring the resulting binary image. Deterministic finite state machine is then used to identify the blink state where a high value of the resulting threshold indicates a blink.

Such methods are not able to localize the eyelids shape. In addition they do not deal with intermediate states where the eye is half closed/ half open.

1.4.2.3 Motion-based methods

Motion-based methods detect blinking by detecting the existence of motion in the eye region from one frame to another.

For instance [GBL+03, GBGB01] employ an eye-motion based blink detector. First the difference image between the current and the previous frame is calculated and binarized. Then, morphological operations, several filters and a stored eye blink motion pattern are used to eliminate noise due to lighting and probable motion due to other moving elements in the image and discard non-relevant candidates. The detected eye is extracted to obtain an open-eye template of the subject in question. Finally, a correlation with this open-eye template permits to determine if the eye is closed or open. This method does not deal with intermediate eye states. In addition it assumes that the first image of the tracked subject is an open-eye image.

Image flow analysis methods were also proposed. For example, [DB08] detects blinking by detecting the motions of the eyes through the use of normal flow. First, a boosted classifier is used to detect the eye region. Then normal flow is calculated in the direction of intensity gradients to track the eyes motion. Using a deterministic state machine, the states of the eye are classified according to the magnitude and direction of the flow.

Such methods may be effective. Nevertheless, relying on motion is not totally robust especially in the presence of other motions of the face. Even if the head motion can be filtered out, however, other local motions during facial expressions are not that easy to be filtered. Therefore, we can state that such methods are robust only if the eyes or facial region is not submitted to any other motions except for that of the eyelids.

1.4.2.4 Parameter-based methods

Parameter-based methods for blink detection are those that tend to encode blinking or eyelids motion in a number of parameters. Typically, these methods might use statistical models such as [Bac09] that builds an Active Appearance Model for the eyes. The model is built on images containing variations in gaze, blink, and pose. The shape parameters are then differentiated to find which modes of variation are responsible to these different actions. Two approaches for blink detection were implemented. The first is based on geometrical distances between the eyelids points. The second is based on identifying the parameter responsible for blinking and then projecting this parameter on the space of blinking parameters extracted from training open and closed eyes on this person. [TCMH11] also uses AAM to track and model the eyelids motion. However, they use Baker et al. formulation [MB04] rather than thats of Cootes. [BCR+07] models the eyelids deformations that are correlated with the eye gaze using trigonometric functions driven by angular parameters. Blinking is modeled using PCA: after labeling high resolu-

tion face examples, PCA is applied and the parameter responsible for blinking is identified and used to control the blink motion.

The parameters of statistical-based methods needs identification in order to figure out which one is responsible for the blink motions. In addition, the detection depends on a database containing eyelids variations. [DOG06] employs the parametric Candide model [Ryd87] to track the eyelids. The eyelids state is encoded in a parameter responsible for this action beforehand. A threshold on the values of the eyelids parameter permits to detect the eyes state.

In this thesis, we present a face model that is able to detect blinking by following the eyelids motions. The presented blink detection algorithm belongs to the parameter-based category. Compared to some of these methods, the proposed approach does not need training on subjects performing different eye closures to be able to follow the eyes motions. It integrates a blinking parameter in the AAM formulation permitting to follow the eyelids of the subject and to give values of the parameter that indicates the degree of opening of these eyelids.

1.5 Conclusion

In this chapter we have reviewed literature concerning face modeling in both synthesis and analysis domains, in addition to literature concerning gaze and blink detection.

We have seen in the face modeling sections the different face models in both the face synthesis and analysis domains, together with underlining the advantages and disadvantages of one over the other. In an attempt to make benefit of the advantages of more than one model, some approaches tend to combine two or more different face models.

Among these approaches, [KHH05] apply a combination of different models to find the different features of the face. They use snakes to fit the lips, deformable templates to fit the eye, k-means clustering combined with snakes to fit the eyebrow. They fit Candide model to the face using the extracted control points. This approach does not deal with big head orientations.

[SUB09] combines 3DMM and AAM to perform robust pose estimation. They build an appearance model out of a set of 3D laser scans instead of annotated 2D images. The fitting procedure is exactly as in the classical AAM. [SK07] incorporate active contours into the AAM fitting to make it independent of the background of the face image. [SKK07] combines AAM and ASMs. They integrate both models into one objective function where the final error to minimize is a combination of the errors of both models. In this way, the two models work together to improve the model fitting. [CPP08] proposes the muscle-based anthropometric AAM. Instead of hand-labeling faces in the learning database, they use a 3D wireframe mesh. The mesh is placed manually on several faces to build the AAM. They parametrize the facial deformation (expression) based on FACS AUs and the muscle model of Waters [Wat87]. To control person specific characteristics, they employ parameters based on anthropometric statistics. They built context-dependent parameters

called the Expression Action Units (EAU).

Combining different models, or applying ideas of one model into another can lead to better performances. For this reason, we present in this thesis an extension of AAM that is inspired from some models presented in the above literature survey. We concentrate our model on gaze and blink detection.

In the following chapter, first we detail the formulation of AAM, then we state their limitations. The chapter is followed by a proposition of a new model that extends the AAM and makes it more robust to certain limitations by inspiring from ideas of other modeling techniques reviewed in this chapter.

Chapter 2

Active Appearance Models: Formulation and Limitations

Sommaire

Active appearance models, originally proposed by [CET98b], are statistical deformable models of shape and appearance made starting from several examples. They can be used to model any object of known morphology. Particularly, they have been widely used for modeling faces.

Generally, AAMs are constituted of two phases (illustrated in figure 2.1), a learning phase (phase of model creation) and a searching phase (phase of model fitting). In the learning phase, a model is built using the variations between different shapes and textures of several learning examples. At the end of this phase, each face in the learning database

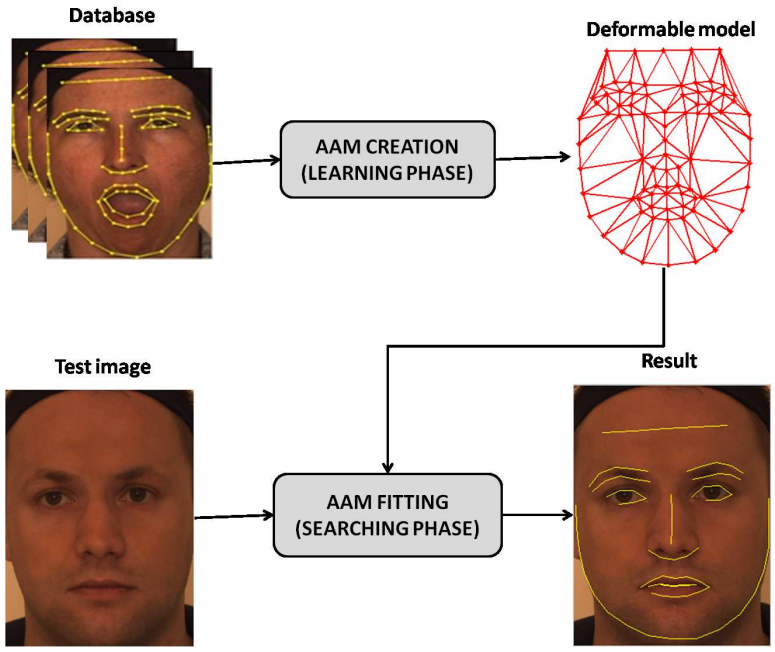


Figure 2.1: Active Appearance Models steps

is described using a set of "appearance parameters". These parameters are capable of regenerating the shape and the texture, and consequently, a near-photorealistic image of each of the learning examples.

In the searching phase, a search of the parameters that best describe an unknown face takes place. For this to occur, a minimization of the difference between the active model and the real image is done through an optimization algorithm.

In the following, we present the basic formulation of active appearance models. Then, we state its limitations together with the extensions we find the most relevant to our work. We Finally discuss how we inspire from the state-of-art models and the synthesis field to suggest an extension of Active Appearance Models.

2.1 Active Appearance Model creation

The AAM creation phase is constituted of four major steps. These steps are explained in the following, and figure 2.2 illustrates them.

2.1.1 Shape modeling

The first step of the model creation phase (also called the learning phase) is the collection of an adequate learning dataset. The choice of the appropriate learning images plays a

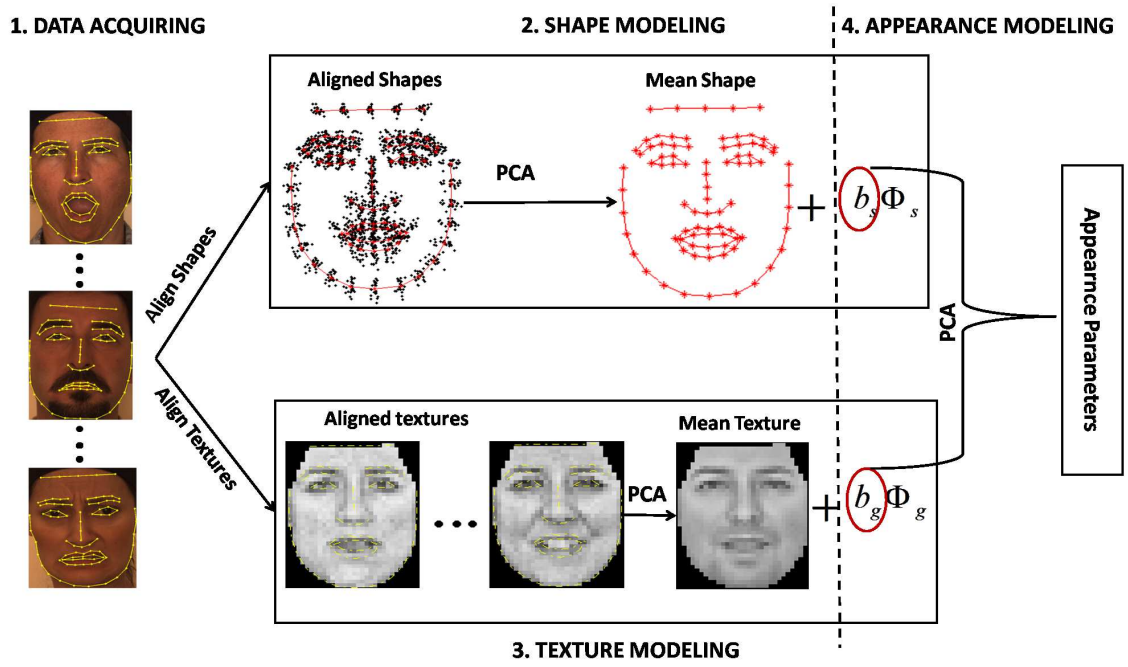


Figure 2.2: AAM creation–Learning process of Active Appearance models

major role in the quality of the searching results. The more variation the learning database contains the more the model will be able to follow the facial variations that might be present in the unknown faces to be searched. For example if we want the AAM to be able to follow a person's gaze, the learning database should contain people with different gaze direction. The model would then learn these variations and thus will be able to follow them in a new image.

After having chosen the learning database, a second step is the landmark positioning. It is the most cumbersome of the phases of active appearance model building. This step constitutes of the placement of definition points on the faces present in the database. These points are put in a way to highlight the borders of the different features of the face (eyes, nose, mouth and chin). One remark is that landmark placement should be done in similar order between the different subjects of the database to ensure correspondence between different shapes.

Landmark positioning might be $2D$ or $3D$. In their original form, AAMs are 2-dimensional. However, extensions to AAMs have used $3D$ landmarks instead [SALGS07b]. In this thesis we use the $2.5D$ AAMs of [SALGS07b], thus $3D$ landmarks. Examples of face annotations is shown on figure 2.1.

Shape alignment –The set of landmarks of a face constitute its shape. Let S_{image} be

the shape vector:

$$S_{image} = [x_1, \ldots, x_n, y_1, \ldots, y_n, z_1, \ldots, z_n] \tag{2.1}$$

where n is the number of landmarks.

After the collection of the different shapes, Procrustes analysis [Ros04] is performed in order to align them. This step aims to filter all the scale, translational and rotational effects from the different shapes by aligning them iteratively on their mean. A point to point correspondence is thus achieved between the different shapes. This alignment results in the calculation of the mean shape $\bar{s}$ and the set of aligned shapes s_i, where i is the image number.

Shape variation modeling –To model the shape variation between the aligned shapes, Principle Component Analysis is performed [Jol05, Shl05] on them. Thus, each shape in the learning base can be described as a linear combination of some modes of variation.

$$s_i = \bar{s} + \phi_s b_s \tag{2.2}$$

where ϕ_s is the matrix of eigenvectors of the covariance matrix of all shapes. b_s is the vector of shape parameters. According to the above equation, shape instances can be generated by modifying shape parameters.

$3D$ **pose parameters** – Using $3D$ shapes, $2.5D$ AAM succeed at synthesizing different head poses without the obligation of learning such variation. In other words, there is no need to learn on people with different poses in order to align faces with different poses. In this way, the appearance parameters of AAM do not include information about the head pose of the person. It is only confined with the color and shape of the face in frontal view. The pose vector capable of manipulating the head pose is then:

$$T = [scale_x, scale_y, \theta_{xy}, \theta_{yz}, \theta_{xz}, t_x, t_y]^T \tag{2.3}$$

where $scale_x$ and $scale_y$ are the horizontal and vertical magnification of the model. θ_{xy} is the rotation around the z-axis (head doing circular rotations), θ_{yz} is the face rotation around the x-axis (head shook up and down) and θ_{xz} is the rotation around the y-axis (head turned to profile views). t_x and t_y represent the translation parameters from the supposed origin.

2.1.2 Texture modeling

Texture alignment – Aligning textures of the learning database means that, for every image in the database, to map the pixels under the corresponding shape into one reference shape. This reference shape is the mean shape calculated from the alignment of the shapes in the previous steps. This operation is called image warping [GM98]. The result is a set of shape free patches, all of the same size. Thus an image texture can be expressed by the following vector:

$$g_{image} = [g_1, g_2, \ldots, g_m] \tag{2.4}$$

where m is the number of pixels in the texture.

Texture normalization – This step serves for actualizing illumination effects between the different textures of the learning database. Many methods were proposed in the literature to tackle this issue [SKM04]. However, we employ the method used by Cootes. Let g_{image} be the original texture and $g_{image_{normalized}}$ be the normalized one, then:

$$g_{image_{normalized}} = \frac{g_{image} - \mu_{image}}{\sigma_{image}} \tag{2.5}$$

where $\sigma_{image} = \sqrt{\sum_{i=1}^{m} g_{image}(i) - \mu_{image}}$ is the standard deviation of the image and $\mu_{image} = \overline{g_{image}}$ is the mean of the pixels of the texture.

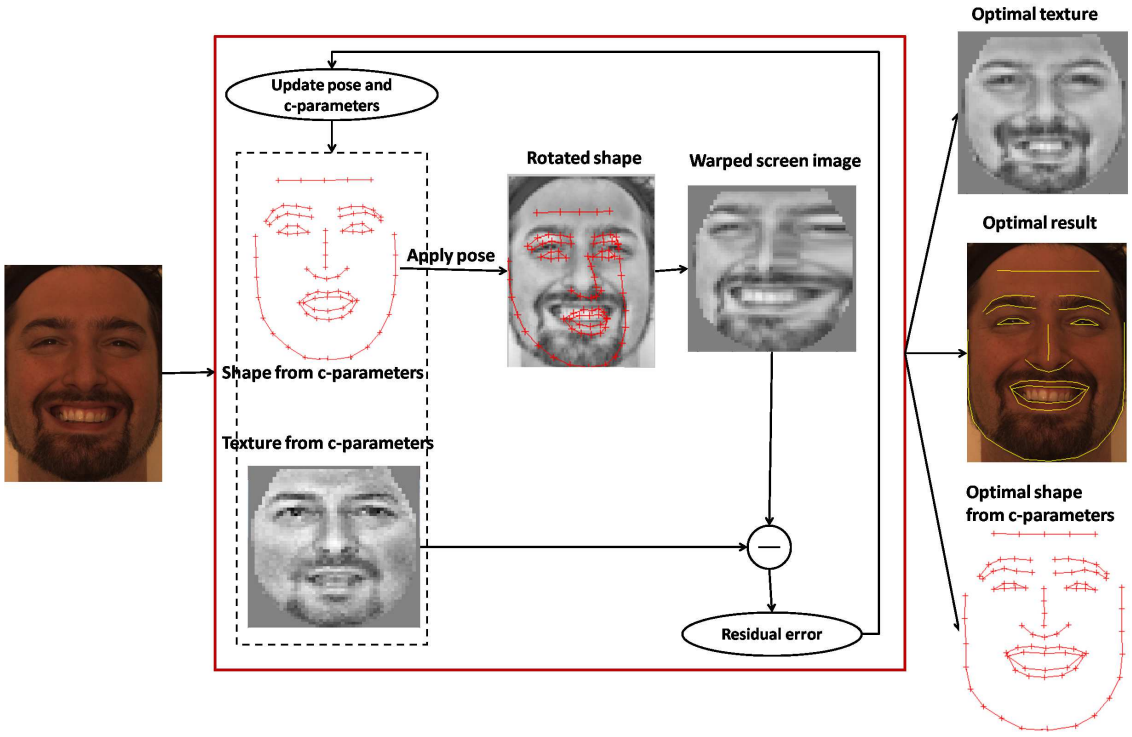


Figure 2.3: Active Appearance Models Fitting

Texture variation modeling – Modeling the variation of textures is done in a similar way to that of the shapes by applying principle component analysis. Any texture g_i can be represented as a linear combination of some modes of variation.

$$g = \bar{g} + \phi_g b_g \tag{2.6}$$

where ϕ_g is the matrix of eigenvectors of the covariance matrix of all the textures. b_g is the vector of texture parameters. Synthesis of the different textures can be controlled by these parameters.

2.1.3 Appearance modeling

Shape and texture models combination – To combine the two models of texture and shape, a third PCA is applied to the concatenation of the texture and shape parameters b_g and b_s. In order to account for the difference between the units of b_s and b_g where the first is in pixel distances and the second is in pixel intensities, a weighting factor W_s is multiplied to b_s.

$$b = \begin{pmatrix} W_s b_s \\ b_g \end{pmatrix} = \begin{pmatrix} W_s \phi_s^T (s - \bar{s}) \\ \phi_g^T (g - \bar{g}) \end{pmatrix} \tag{2.7}$$

This results in the appearance parameters C capable of manipulating the shape and the texture simultaneously.

$$b = \phi_C C \tag{2.8}$$

Shape and texture can be found using the c-parameters by the following formulas:

$$s = \bar{s} + V_s C \tag{2.9}$$

$$g = \bar{g} + V_g C \tag{2.10}$$

where $V_s = \phi_s W_s^{-1} \phi_{C,s}$, $V_g = \phi_g \phi_{C,g}$ and $\phi_C = \begin{pmatrix} \phi_{C,s} \\ \phi_{C,g} \end{pmatrix}$.

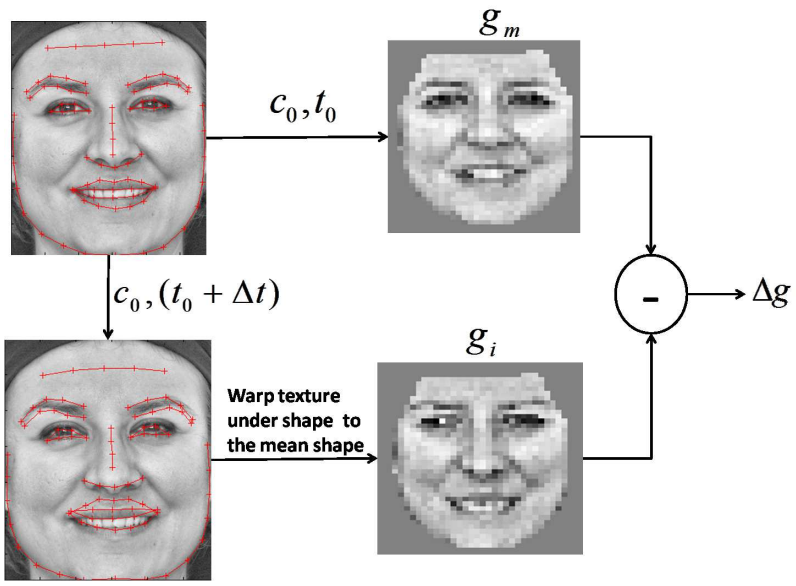


Figure 2.4: AAM training process

2.2 AAM fitting

In the searching phase, given a new face image, the aim is to find the pose and appearance parameters that best describe the shape and appearance of the face image in question. The model parameters are varied to synthesize new instances of face image. The problem is an optimization problem where we opt to find the optimal vector of parameters that best minimizes the pixel error between the synthesized facial image by the model and the real one.

$$\Delta g = g_{image} - g_{model}(C, T) \tag{2.11}$$

$$E = |\Delta g| \tag{2.12}$$

where g_{image} is the texture vector corresponding to the image, and g_{model} is the texture vector synthesized by the model. Δg is the difference between the two and is called the residual error. E is the pixel error, defined to be the norm of the residual error. The optimization of this error is classically perfomed using simple gradient descent algorithm.

Training – In his original formulation, to solve the optimization problem of the search, Cootes proposed to learn a relationship between the model parameters' variation and the residual error between the model and the image. Actually, each face image of the learning database is interpreted using a set of appearance parameters during the learning phase and thus can be synthesized using these parameters. Applying small perturbations to these parameters and calculating the corresponding residual error permits to learn how to tune the parameters in the searching phase in a way to drive the model towards the optimum.

$$\Delta C = R_C \Delta g \tag{2.13}$$

$$\Delta T = R_T \Delta g \tag{2.14}$$

where ΔC and ΔT are the perturbations to the model parameters, Δg is the corresponding residual error. R_C and R_T are the appearance and pose regression matrices respectively.These are calculated according to the following equation. The mathematical derivation is found in [CT04].

$$R = \left(\frac{\Delta g}{\Delta p}^T \frac{\Delta g}{\Delta p} \right)^{-1} \frac{\Delta g}{\Delta p}^T \tag{2.15}$$

where p is the parameter in question, it can be either C or T. A glance of the training process at one iteration is shown in figure 2.4.

AAM search –Having the regression matrices calculated during the training phase, the search is done using these matrices. Figure 2.3 depicts the steps of AAM search. These are presented in the following:

– Initialize the model parameters C_0 and T_0 and set the number of iterations

While Number of iterations not reached **do**

1. Calculate the initial residual error Δg_0

2. Predict displacements in c and t (ΔC and ΔT) according to equations 2.13 and 2.13 respectively

3. for k = $[1, 0.5, 0.25]$ called the damping factor
   - Predict a new value of C and t: $C_{k_i} = C_0 - k_i \Delta C$, $T_{k_i} = T_0 - k_i \Delta T$
   - Generate the model shape s_{model} and texture g_{model} using equations 2.2 and 2.6 respectively
   - Apply the pose vector on the shape coming from the model to obtain the shape on image s_{image}
   - Map the pixels under s_{image} into the mean shape to form g_{image}
   - Calculate the corresponding residual Δg_{k_i} using equation 2.11
   - Calculate the corresponding error E_{k_i} using equation 2.12

4. Update C and T by the one that gives the least error among C_{k_i} and T_{k_i}

Now that we have detailed the formulation of AAM, we move to specifying the limitations in this classical model in the following section.

2.3 Challenges, Limitations and extensions of AAMs

AAMs have proved to be very efficient in modeling faces. However, they still present several drawbacks. To tackle the following presented limitations, researchers have proposed extensions to the AAM initial formulation.

The limitations of AAM concern the initialization of the model, its background, the search algorithm, its generalization capabilities which include its robustness to different factors (such as illumination variation, occlusions, head pose,...) and finally the fact that it produces a set of non-decoupled parameters. Concerning initialization and background problems, the work in this thesis does not deal with these, so we will not review nor discuss related literature. Moreover, to initialize our model, we use the popular Viola and Jones face detector [VJ04].

The work in this thesis can be classed in the category of models that deal with generalization and non-decoupled parameters limitations of AAM. For this reason, we place our literature review in these areas. To increase the generalization capabilities of AAM, some methods tackle the search algorithm. Thus, we additionally review some methods concerning this limitation. Figure 2.5 presents the AAM limitations together with the state-of-art solutions to these.

In the following, we present this review and we position our research with respect to the state of the art.

For a complete overview of the AAM extensions literature, [GSLT10] is a recent review that studies all of the aspects of these.

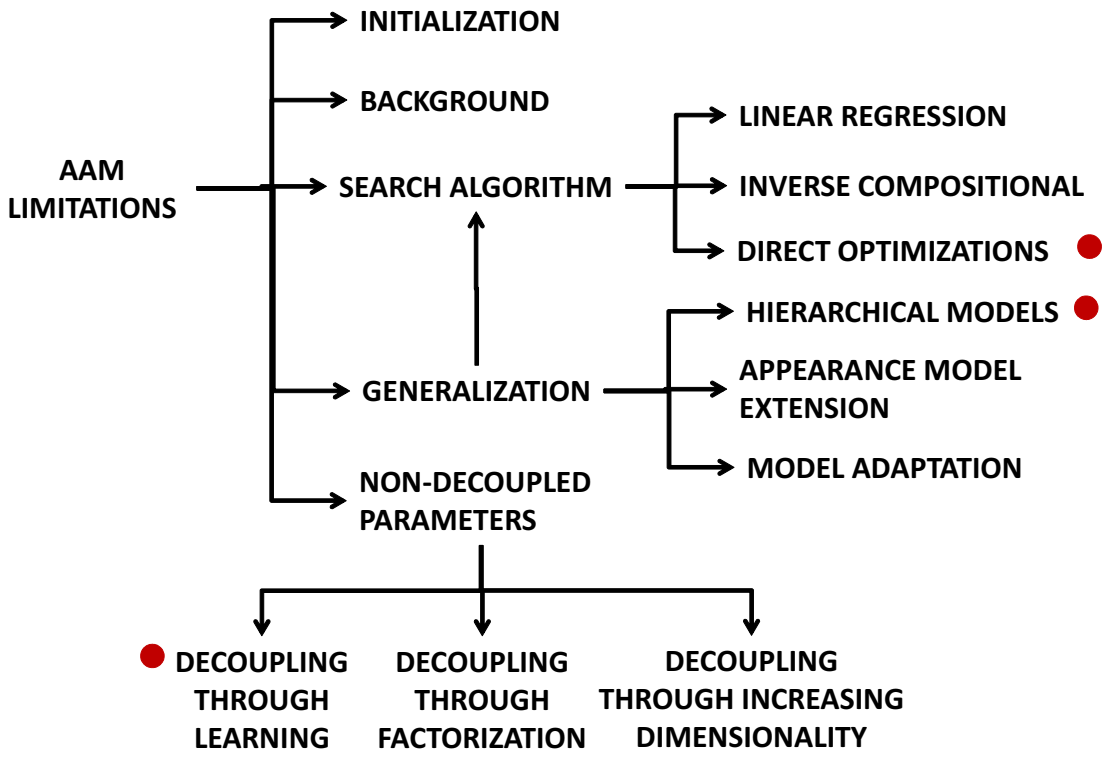


Figure 2.5: Limitations of AAM together with some of the methods usually used to solve these limitations

2.3.1 Generalization

One of the biggest challenges in AAM fitting is to be able to generalize to new examples. This means the ability to give accurate fitting results for faces that are not present in the training database. For an AAM to be able to generalize to new faces, the training database should include all the possible variations that might be present in the new image. This includes the variations due to head pose, facial expression, lighting and person-specific characteristics such as age, color, and ethnicity, in addition to the presence, or not, of facial hair (beard, buck or hair on the forehead). Even the quality and resolution of the training images affects fitting new images.

However, generalizing to new examples is not that easy to reach. In fact, the naive solution is to include all the possible variations in the database. Unfortunately, this presents some disadvantages. First, increasing the size of the training database will result in increasing the number of appearance parameters. This makes the segmentation phase of AAM more computationally expensive making the models non convenient to a real

time implementation. Second, a learning database containing a many variations can lead to a parameter space containing classes (clusters) corresponding to the different present variations. The more the parameter space is clustered, the more gaps will be present between these clusters [SLGBG09]. This makes the convergence of the AAM a more difficult task.

Methods that try to increase the generalization capabilities of the AAM are many. We will not review all of them, however, we mention those that extend the appearance model (section 2.3.1.1), those that adapt the AAM to the person (section 2.3.1.2), those that extend the search algorithm (section 2.3.1.3) and finally those that use a hierarchical representation of the face (section 2.3.1.4).

2.3.1.1 Appearance model extension

The texture in AAM is represented using pixel intensities. This is prone to influence by appearance variations and thus might affect the fitting results. To improve the generalization capabilities of AAM, and thus obtaining better fitting performances on unseen images, some methods extend the appearance model of AAM. Intensity is replaced by another representation. For instance, [TAiMZP12] propose the Active Orientation Models (AOMs). In this approach, the authors build the appearance model based on images gradient orientation instead of gray-scale image intensities. The authors claim to generalize better to unseen faces with respect to the the classical AAM. Figure 2.6 shows a fitting of a face using the AOMs together with the texture representation using the gradient orientation maps. A bunch of approaches use Gabor filters as a representation of the

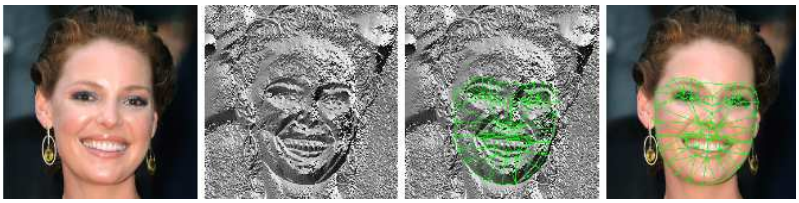


Figure 2.6: Gradient orientation maps of [TAiMZP12] for appearance representation

image texture [YDJ+13, GSLT09, SGTL08]. Such representation takes into consideration local structures of the image [GSLT09]. [GSLT09, SGTL08] directly use the Gabor magnitude and phase for the texture representation. The image intensity texture of AAM is convoluted by Gabor filters. On the other hand, [YDJ+13] propose to use the statistical characteristics of the Gabor magnitude and phase. According to these characteristics, different texture representations are presented and compared. The authors claim that the advantage of using the statistics of the Gabor filters over directly using the magnitude and phase as in [GSLT09, SGTL08] is that it is computationally less expensive since they reduce the dimension of the texture vector of AAM with respect to the formers.

Manipulating the texture representation seems to improve the fitting results of AAM. However, this increases the computation time of AAM since more operations are applied on the image before collecting the texture information. In this thesis, we do not use such approach. We prefer to tackle other sides of the AAM to improve it.

2.3.1.2 Model adaptation

On the other hand, other methods propose to adapt the model to the new face. [SLGBG09] propose the adapted AAM. In this approach, the authors first build a general identity AAM. For an unknown face, the distance between the parameters of this face from those corresponding to the faces in the identity database is measured. According to this measure, a model is adapted for this face by building the model on the expressions of the faces that were found to be the nearest to the face in question. Although such scheme is efficient, however it still needs the availability of a large database that includes wide variabilities. In addition it might be possible that even with the most adapted model, the results of search wont be accurate. [CK04] adapt the AAM to the identity of the subject. First they apply an AAM on the neutral face to find the optimal facial features and parameters. A 3D face model is then constructed using the aligned face by adapting the resulting texture on a 3D face model. Different views of this model are then used to construct an AAM which is adapted to the variability in identity and in pose. This method is dependent of the alignment results of the neutral face in addition to the constructed model.

2.3.1.3 Search algorithm extension

One of the reasons of the efficiency of the AAM in terms of speed is its search algorithm. The linear assumption between the texture error and the parameters updates bypassed the need to deal with a high dimensional non-linear optimization problem, minimizing by that the calculation time. Indeed, this whole search scheme including the training phase was one of the novelties that Cootes had brought to the community. However, it presents certain inconveniences. These come basically from the linear assumption between the texture error and the parameters updates which was proven to be untrue by [MB04]. When this assumption fails, convergence of the AAM also fails. This is why many alternatives and extensions were proposed in the literature. We classify these into three classes: Linear regression methods, inverse compositional methods and direct search methods.

Linear regression methods

Due to its efficiency, some methods keep the original optimization scheme but try to improve different sides of the algorithm. [CT06] and [BH05] update the constant regression matrix in terms of the search image. [DRL+06] replaces the multivariate regression method by Canonical Correlation Analysis (CCA) assuming that it better models the relation between the residual and the parameters update. [CET98a] computes the regression matrices based on the shape parameters to reduce the calculation time of the training phase and the dimension of the matrices. [SG07] learns the relationship between the

parameter update and the error using non linear boosting.

Inverse compositional methods

Other approaches choose to completely replace the original optimization scheme of AAM. [MB04, PM08] used the inverse compositional approach. They reformulate the problem as an image alignment problem by relating it to the Lucas-Kanade algorithm [LK81]. In this approach the model parameters are updated in function of the residual image, the steepest descent images, and the Hessian matrix.

Direct optimization methods

Direct optimization methods such as Nedler Mead Simplex algorithm [NM64] and Genetic Algorithm (GA) [Rec71] were also used. These methods are known for their exploration capabilities (ability to span all the search space of parameters). For instance, [SALGS07b] used the simplex to fit the $2.5D$ AAM. This makes AAM more efficient in terms of memory consumption and thus more suitable for embedded systems. On the other hand, [SAS08, SMG09] employed the GA in their face alignment search. To decrease the computation time which is very high, [SAS08] employed a Gaussian mixture clustering of AAM. [SS10] hybridized simplex and GA to improve the AAM search to obtain even more robust and accurate fitting.

Even though such methods take longer time to converge, however their ability to reach optimal results is bigger than classical linear regression methods. The reason is that the linear regression method updates the parameters according to a non-totally true assumption of linearity between the model parameters and the texture. Thus, when this assumption breaks during the search of AAM, the direct methods will perform better since they do not base their search on any *a priori* assumption. Compared to the classical gradient descent algorithm, the latter exploits the current solution and converges toward the negative of the gradient but do not go further. However, these methods expand the search space and explore every other solutions.

In this work, we use methods from the direct search paradigm to optimize some parameters of our model. We further prove that direct methods work better for us through several experimentations.

2.3.1.4 Hierarchical models

Instead of using one global model to search the face, multiple local models for different facial areas (mouth, eyes, nose or upper part and lower part of the face) could be combined together or with the global model to make the search.

[ZC05] proposed the component based AAM. In this approach, iteratively, the sub-models update component points independently then they are united to a global AAM. [Bac09] also uses a similar approach to model the eyes. [RCA03] uses a global model to apply iteratively-updated soft constraints on a sequence of partially overlapping sub-models. [ZG05] models the face by a two-level hierarchical person-specific model. The first level accounts for the component facial features (mouth and eyes). The second one combines the facial sub-models to model the final expression using expression variabilities

of sub-components. [PBMD07] proposes the multi-level segmented AAMs. Each segment encodes a component of the face. A coarse-to-fine fitting strategy which gradually splits AAM into pre-defined increasing number of segments is used which increases the convergence. [XCZL08] proposes the hierarchical-compositional model to model the face. The face is represented by three layers. The first layer deals with the face as a whole entity. The second one deals with the components constituting the face (eyes, eyebrows, nose and mouth) and refines the alignment of these components using a set of individual templates. The third one deals with the fine details such as wrinkles. They argue that transitions from coarse to final layers leads to structural changes translated by the facial deformations. They deal with these transitions using And-Or graphs. [MCRH06] also employs a hierarchical model of appearance. The major appearance modes of each sub-facial model encode the major variation for the corresponding area. For example, the highest mode of variation in the left-eye model encodes a blink.

Through the use of PCA, AAM supposes a linear relationship between the different parts of the object in question. This assumption is not always valid. Non-linearities of AAM might occur for example in cases of articulated object where some of its parts move independently. [LPDB05] propose a multi-features method to deal with non linear shape variations of AAM. They use the Minimum Description Length (MDL) to optimally identify independent distinct entities of the face called "cliques". The MDL automatically extracts the cliques from the space of proper values of the learning base of aligned shapes. Each of these entities is then modeled using AAM. Finally, a global AAM is created by regrouping the close cliques two-by-two. Even though this method is efficient for dealing with the non-linearities of AAM, it is not robust for large head poses.

In this thesis, and due to the efficiency of hierarchical modeling of the face in giving accurate results for facial features, we use such representation in our modeling. Actually, we use it in two stages of our work. First in our proposed face model described in chapter 3 and second in the work presented in chapter 4 which was in the context of a grand challenge for emotion recognition. A detailed description of how this representation was used in our work is done in these chapters.

2.3.2 Non-decoupled parameters

One of the problems of standard AAM is that the parameters it gives are not easily decoupled. Since PCA jointly models all the variations in the database, AAM gives a set of parameters that are not well-identifiable. Being able to decouple the different parameters into interpretable ones, or to assign each variation a specific parameter is an important matter. Actually, this permits to directly use the information in different contexts of applications. In addition, using a non-decoupled model can introduce non-valid spaces and generate non-realistic shape/appearance configurations.

Approaches in the literature try to deal with this issue in different manners. These methods tend to subtract information from the appearance parameters by assigning certain variabilities of the face (for example, pose, identity or illumination) to a certain number

of parameters other than those in the appearance ones. Thus, the appearance parameters describe other variabilities than these entities and the parameters modeling these variations are optimized in parallel with the appearance parameters. Decoupling can be done using factorization, increasing dimensionality or learning.

2.3.2.1 Decoupling through factorization

Some methods tend to factorize the appearance space into several sub-spaces, each responsible for a certain variation. Bilinear models were used to decouple variations in pose, identity and expression.

Decoupling pose– [GMDlTM+07] decouples the pose appearance space from the expression/identity space through asymmetric Bilinear AAM (BAAM). Bilinear models are two-factor models with the property that their outputs are linear in either factors when the other is held constant. They provide rich factor interactions by allowing factors to modulate each other's contributions multiplicatively [AD05]. Through this BAAM, different poses are possible to be synthesized using a number of parameters that control the pose space.

Decoupling expression – [AD05] also uses bilinear models to separate the expression and identity factors from the appearance parameters of AAM. The appearance model is built starting from a learning database which is made using a set of neutral and expressive faces. They compare the performance of asymmetric and symmetric bilinear models for expression synthesis.

Even though this approach efficiently decouples variations from the appearance parameters of AAM in a way that such variations are independently controllable, however this decoupling is done *a posteriori*. In other words, such technique does not surpass the need of a large database containing large variability. In this thesis, we are interested in subtracting variability from the AAM parameters together with restricting the AAM learning database. We wish to extrinsically model certain facial deformations. On the other hand, it is not obvious if factorization can be used for decoupling facial actions and not expressions.

2.3.2.2 Decoupling through increasing dimensionality

In order to decouple the variations in pose from the appearance parameters, some methods propose to increase the dimensionality of AAMs. This avoids the necessity of including subjects performing head rotations in the learning database and permits to construct a sub-space of parameters that controls the variations in pose.

Decoupling pose – [SALGS07b] proposes to combine a $3D$ shape and a $2D$ texture to form the $2.5D$ AAM. The third dimension of the shape corresponds to the depth information which is calculated by annotating profile views of the subject's face. This way, the appearance model is built on frontal-view subjects and pose information is extracted from the appearance parameters. [XBMK04] also uses $3D$ modeling of the shape. The

authors prove that $2D$ AAMs are able to model the same phenomena as $3D$ models but with a larger number of parameters. They propose the $2D + 3D$ AAM. First they use a non-rigid structure-from-motion algorithm to construct $3D$ shape modes that correspond to $2D$ AAM. Then the $3D$ modes are used to constrain the AAM so that it can only generate model instances that can also be generated with the 3D modes.

Decoupling lighting –[ARARCE11] extends 3D active appearance models to model illumination. In their approach, they separate lighting and texture modeling. They parametrize the appearance due to illumination by using spherical harmonic functions. Their algorithm infers simultaneously the shape, texture, pose and lighting parameters.

Increasing the dimensionality of AAM would only solve the head pose parametrization: no different head pose variations are needed to be included in the AAM learning database. As we are interested in this criterion, we choose to adopt the $2.5D$ model of [SALGS07b]. This model uses 3 dimensional shapes with 2 dimensional textures. This subtracts pose variations from the learning database but keeps low computational processing since there is no need for 3D texture computation. However, increasing the dimensionality of AAM is not sufficient to decouple all of the facial variations. For this reason another approach should be integrated to the $2.5D$ representation.

2.3.2.3 Decoupling through learning

Some methods decouple some particular variations from the appearance parameters through the construction of specialized databases.

Decoupling illumination variation– In [KS06], the authors propose to construct a database containing subjects each acquired under a certain number of illuminations. Variation of illumination is done by varying the position of the illumination source. Next, the illumination variation is modeled using Principle Component Analysis (PCA). It was noticed that the first variation mode describes identity and that the second describes illumination variation.

Decoupling identity and expression – [CTC10] build two separate 3D shape models, one for identity and one for expression. The identity model is built from training data containing people with a neutral expression, eyes open, and mouth closed. The expression one is built from a small set of facial actions created from a neutral base. Identification of the modes of variation is then needed to determine which mode is responsible to what facial action. They present two approaches to fit the two models. The first one combines the parameters of both models into one, and then the fitting procedure is done similar to fitting one model. The second approach fits the two models in an alternating process. This is done by substituting the results from the ID model into the actions model, and vice-versa. Therefore, at each iteration of the algorithm, both models are fitted in sequence to the same target before moving on to the next iteration. [CTC12] extends the method of [CTC10] by employing a hierarchical representation. The parts of the face are fit independently which results in an improvement of the fitting results.

Building a specific database that models a specific variation is an attractive approach.

In the next chapter, we present a face model that uses ideas from this class of methods. The objective of our model is to decouple variations from the appearance parameters of AAM. To parameterize the motion of the eyeballs for example, we specialize a database that models the appearance of the iris. This appearance however has nothing to do with the eyeball motion and only concerns the color of the iris. Our model thus differs from the state of art in this aspect. More details are presented in the dedicated chapter (cf. chapter 3).

2.4 Conclusion

We have presented, in the above, the formulation of Active Appearance Models. We have also underlined the limitations and challenges of the standard AAMs. We have focused on two limitations: the generalization capabilities of AAM and the fact that the parameters produced by AAM are non-decoupled and well-defined.

In order for an AAM to be able to generalize to new faces and to produce accurate fitting results, a large database containing wide variations should be constructed. This increases the number of appearance parameters making the AAM non-suitable for real time applications. Generalization capabilities of AAM have been tackled in the literature using: Appearance model extension, Model adaptation, Search algorithm extension and Hierarchical models. Among these, the hierarchical modeling of the face leads to more accurate results in specialized regions such as the eyes and mouth. We thus present a model that uses such representation.

On the other hand, compared to the models existing in the literature (reviewed in the past chapters) such as Candide model and parameter-based models used in the synthesis domain, AAM does not produce identifiable parameters. Nevertheless, AAM is based on real data.

Our idea in this thesis is to propose a face model that decouples some facial actions from the appearance parameters of AAM. The model deals with the different parts of the face as different objects which can be related to hierarchical models where each region of the face is assigned a separate model. A set of parameters are added to the AAM formulation. These are responsible for some eye action such as the gaze and the blink. This model restricts the learning database of AAM to little variations concerning identity for example making the model more suited from real-time applications. The proposed model is able to generalize better to new faces with respect to the classical AAM. The ultimate goal is a model made starting from a training database that contains expressionless subjects but is able to track people with different expressions and head pose. Details on our proposition are presented in the next chapter.

Chapter 3

A multi-object facial actions AAM

Sommaire

When a person sees another person, he directly recognizes his different facial features. He can automatically identify their shapes and easily interpret any movement of these. The human mind is used to such identification and interpretation. The objective of a face model is to automatize the identification and the interpretation of facial features and their motion. The more the face model is capable of this interpretation, the more efficient it is to be integrated in real world automatic scenarios.

The goal of this chapter is to introduce a new model of the face. Inspiring from ideas coming from the synthesis domain (see state-of-art methods concerning this domain in chapter 1, section 1.2), we propose the multi-object facial actions AAM optimized using a framework of a multi-objective optimization.

The eyes, being the window to the brain and soul, play an important role in understanding human intentions and emotions. Eye movements in particular are primary cues of non-verbal communication. Through our eyes' gazing and blinking, we can communicate messages to people around us. Through these motions we equally interact, act and react. A face model that is capable of accurately interpreting these motions is an important requirement for automatic face analysis. Accordingly, we orient our face model for the purpose of gaze and blink detection.

The organization of this chapter is as follows. First, we describe the global idea of our proposed model in section 3.1. In sections 3.1.1, 3.1.2 and 3.1.3, we permeate in the details concerning the proposed model. Section 3.2 presents different results concerning the application of the model to the problematics of gaze and blink detection. Finally, in section 3.3 we conclude the chapter.

3.1 Introduction of the proposed model

One of the major disadvantages of statistical Active Appearance Models is the necessity of a big training database that includes all the possible variations in facial expressions and motions so that the model would be able to robustly align the face and accurately find its features. In this thesis, we aim at producing the variations due to certain facial actions without the necessity of including subjects performing these actions in the learning database of AAM. In this way, we subtract from the appearance parameters of AAM the variations corresponding to certain motions of the face. Just like the head pose is modeled separately from the appearance parameters in the $2.5D$ AAM such that there is no need to include faces with different head poses in the learning database in order to find the head pose of a new face image, we want to reach an AAM where facial motions are modeled separately.

We thus propose to extend the pose parameters of AAM. We present a new face model that combines the advantages of the statistical AAM and the interpretable parameters of models such as Candide. This model parametrizes some facial actions which are simple to be parametrized such as the eyebrows, the eyelids and the eyeballs motion, but keeps the AAM statistical learning for more complicated parts of the face such as the mouth or the eyes shape. Figure 3.1 illustrates the idea of the model. It shows the deformable AAM with the different facial actions on it. Section 3.1.1 details this proposition.

In our proposition, the face is represented as an aggregation of different objects. Each object is related to a specific texture. An object can be any facial feature or a combination of facial features. For example it can be the mouth, the left eye, the right one, both eyes or the whole face. The choice of the division of the face into the specific objects

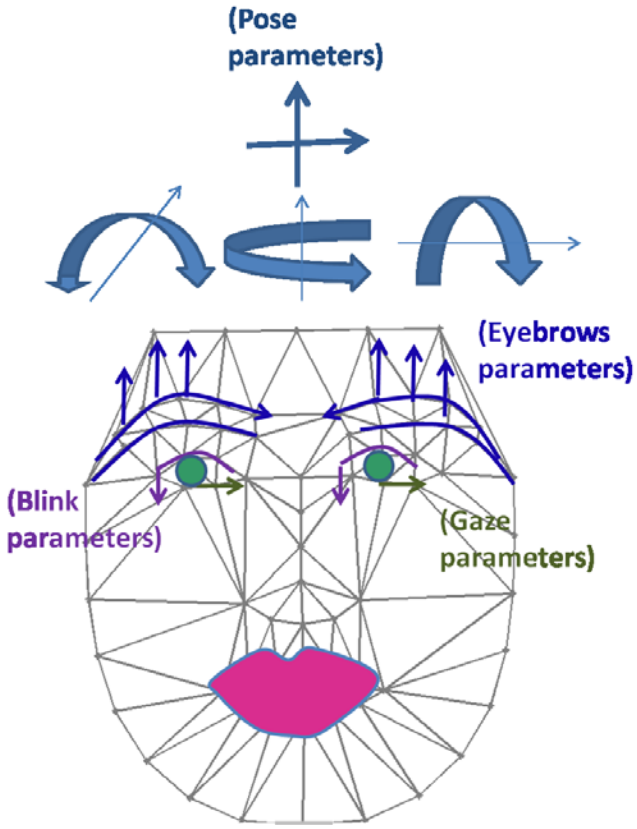


Figure 3.1: Facial actions representation of the face

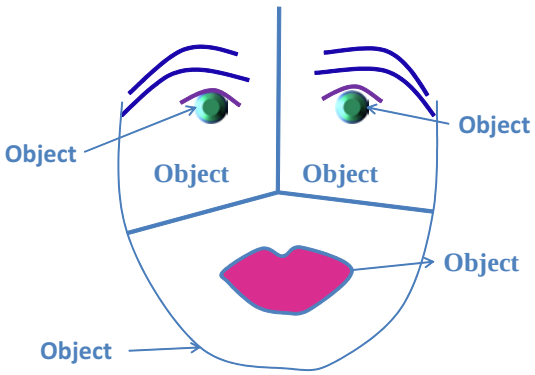


Figure 3.2: Multi-Object representation of the face

depends on the problem in question. Moreover, inspiring from the synthesis domain, one particularity of our model is that we consider that the eyeballs are separate entities from

the face. Figure 3.2 illustrates this representation of the face and section 3.1.2 details it. Separating facial objects contributes in decreasing the size of the database of AAM since the object can be tuned separately. This evades including cases where mutual variations in different features should be included. For example, for a global face to be able to analyze cases where a subject opens his mouth and moves his eyebrows in different positions, the database should contain the combination of the mouth movement and these different eyebrow movements.

As we want to analyze the face in different poses, the information from different objects does not have the same quality. For instance, the left eye can not be analyzed with the same quality as the right one in some cases if the face has a big orientation. Thus, we propose a multi-objective optimization for the proposed model. This optimization takes into account the head pose to favor one or more objects over the others in specific cases. This is detailed in section 3.1.3.

The proposed model includes the following parameters:

$$AAM_{Facialactions} \mid \begin{array}{l} Object(i)\ appearance\ parameters \\ Head\ pose\ parameters \\ gaze\ parameters \\ Eyelids\ parameters \\ Eyebrows\ parameters \end{array}$$

where $Object(i) \in \{Lefteye, Righteye, Mouth, Globalface\}$.

We orient our model principally towards gaze detection. The motivation behind this is attributed to the importance of such analysis for HCI. As a matter of fact, gaze detection enters in wide various applications starting from entertainment interactive applications (virtual reality or video games ([IJSM09])), video conferencing ([Ver99]), aiding disabled people (eye typing or eyes for moving cursors ([VdKS11])) to applications anticipating human behavior understanding through eyes. On the other hand, extracting eye information is not an easy task. This can be summarized by the following:

- The eye changes its states with every blink and with every iris motion;
- It can be occluded by different factors such as eyeglasses and hair;
- It has a small texture with respect to that of the skin surface;
- It is influenced by the variability in the light conditions and in iris location, color and scale.

To tackle these difficulties, we show how our proposed model applied to the gaze detection problematic increases the robustness of gaze detection with respect to classical AAMs (cf. section 3.2).

In the following sections, we detail our proposed model. Section 3.1.1 presents the Facial-Actions AAM. It concerns parameterizing facial motions that are coupled with the skin such as the eyebrows and eyelids motions. Section 3.1.2 presents the Multi-Object

AAM. It concerns dealing with the face as several objects. Section 3.1.3 presents the multi-objective optimization framework used in our work.

3.1.1 Facial Action AAM

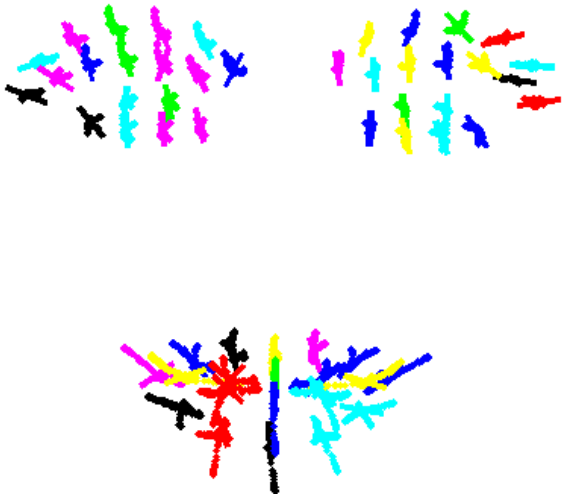
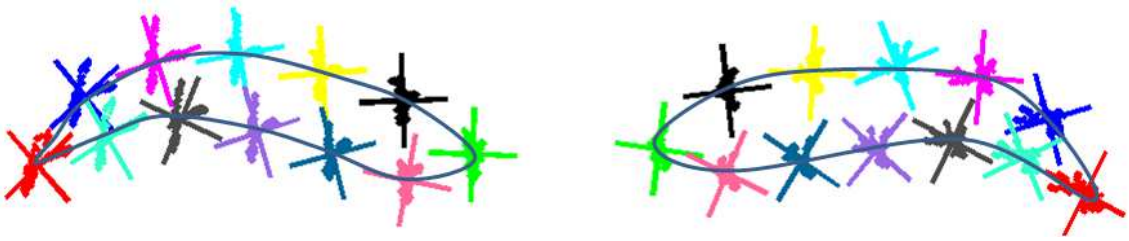


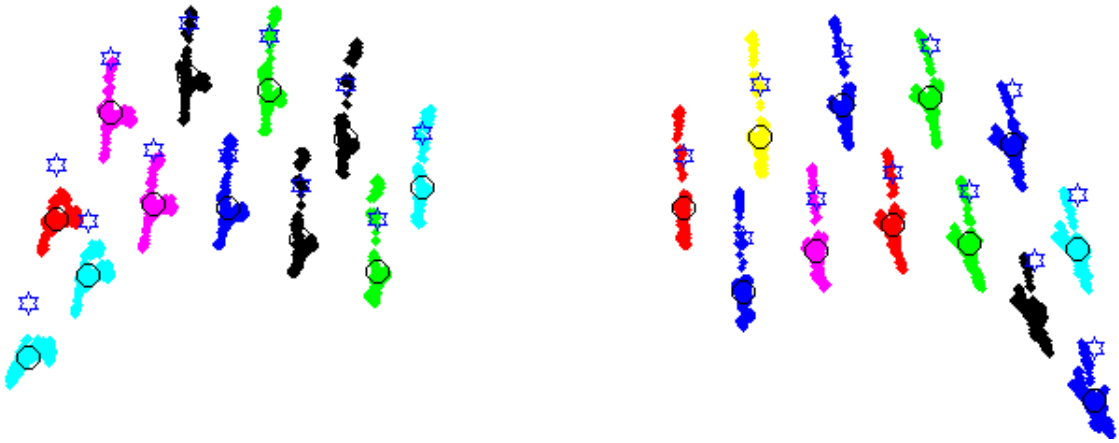
Figure 3.3: Identification of the principle axes of the displacement of the facial landmarks of one subject. These landmarks are traced while the subject blinks his eyes, lowers his eyebrows, frowns and says A/I/O phonemes.

As we have previously said, we are interested in an Active Appearance Model that is capable of generating valid movements of the different parts of the face without the necessity of including such variations in the database. To parameterize the motions of the eyebrows and eyelids, we study the variations of specific local points of the face. In this approach, we have inspired from the method employed by [SBS10] for modeling facial dynamic transitions during an expression for the purpose of facial animation. The authors were interested in observing short term dynamics controlling the transition from one expression to another. So they have collected a $2D$ database of varied dynamic emotional facial expressions performed by an actor and tracked the movement of a set of markers over time.

Hence, we have took a similar path to analyze the movements of eyelids, eyebrows and the mouth. We have asked 10 subjects to perform a number of specific actions with their eyelids, eyebrows and mouths. These actions are constituted of: 2 consecutive blinks, eyebrows up and down, frown, and the pronunciation of three letters to simulate three mouth movements (”A” to simulate open mouth, ”O” to simulate round-shape mouth and ”E” to simulate a smile). The videos were then annotated and global head motion was filtered using Procrustes Analysis. Next, the trajectories of each of the landmarks during the before-mentioned facial actions were observed. Figure 3.3 shows the landmarks



(a) Landmark variations for the left and right eyebrows



(b) Simulating a new position of the eyebrows of another subject by changing the eyebrows parameters. The black circles correspond to the mean points and the blue stars are the new generated points. We notice how the generated points fall into the cloud of points of the subject when he is really performing eyebrows movements.

Figure 3.4: Variation of the landmarks for the left and right eyebrows of one subject during the eyebrow motions: eyebrows up/down and frown. The principle components of every landmark are overlaid over the cloud of points of this landmark.

of a subject while performing facial motion. As we see, the visual trajectories of the markers are organized around well-defined axes. To find these axes we perform PCA on each set of points for each landmark.

Having the principle motion components identified, the motion of a new point can be simulated in the direction of the most significant principle component (PC) or in the direction of both PCs (having two dimensions for the landmarks, we get maximum two PCs). The idea is to assign each group of feature points significant parameters that serve as controllers of these group of points. For instance the motion of the eyelids or the eyebrows. These feature points will be moved in the direction of each of the found principle components.

For instance, concerning the eyebrows, we notice that the PCs of each of the landmark points are similar. Figure 3.4 shows the variation of each of the landmarks of the eyebrows

for one subject while performing the eyebrows movements. It is clear from the figure the resemblance of the PCs of these landmarks. In other words, we can notice that all the landmarks of the eyebrows move in similar directions when performing up/down movements or frowning. For this reason, we approximate the PCs of each of the landmarks of the eyebrows by those of one landmark. These PCs identify two parameters responsible for the eyebrows motions. The first parameter is responsible for the up/down action and the other parameter is responsible for the frown action. Let T^{EB_L} and T^{EB_R} be the action vector containing the parameters responsible for the left and right eyebrows movements respectively. The eyebrows parameters become:

$$T^{EB_L}\phi_L = [T^L_{hor} T^L_{ver}]\phi_L \tag{3.1}$$

$$T^{EB_R}\phi_R = [T^R_{hor} T^R_{ver}]\phi_R \tag{3.2}$$

where ϕ_L and ϕ_R contain the eigenvectors of the covariance matrix of the specific landmarks we chose of the left and right eyebrows respectively.

Concerning the eyelids motion. Two actions are possible: winking and blinking. However, blinking is more frequent. As a starter and to stay simple, we assign only one parameter for both eyes, thus winking is not possible using our model.

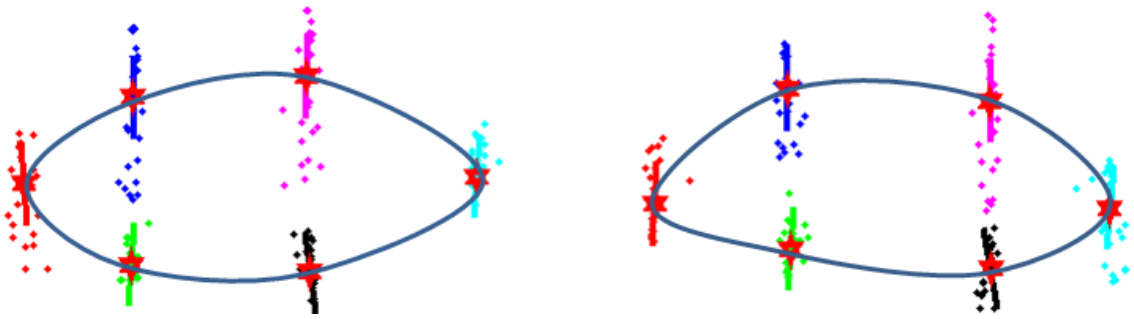


Figure 3.5: Variation of the landmarks for the left and right eyes of one subject during blinking. The principle components of every landmark are overlaid over the cloud of points of each of these landmarks. The red stars represent the mean of each of the landmarks.

The landmark motion analysis of figure 3.5 shows that during blinking, both the upper and lower eyelids move in addition to the corner points. However the motions of the lower and the corner points can be considered negligible with respect to the upper ones. In addition, [TCMH11] proved in their experiments that the lower eyelids motion has no effect on the results of animating characters. For this reason, we assume that during blinking, only the upper eyelids move in our model. Let T_{Blink} be the parameter responsible for the blinking motion. Figure 3.5 also shows that the principle directions of the upper landmarks are vertical. Thus, it is sufficient to add the blinking parameter to the upper eyelid points in the vertical direction.

Integrating the blinking parameter in the AAM considers only the searching phase. Actually, the goal is to restrict the learning base of AAM so that it would be able to detect different eye states without the necessity of including this variation in the learning database. Thus, training is done on a set of open-eyed images. The pose and appearance parameters are trained normally using regression matrices. Then a blinking parameter is tuned during the searching phase of AAM. This blinking parameter is normalized between 0 and 1. The 0 value corresponds to an open eye whereas the 1 value corresponds to a closed eye. Values between 0 and 1 correspond to intermediate eyelid closures.

Since the search space of the blink parameter is considerably small, exhaustive search seems convenient. Exhaustive search, also called brute-force search consists of checking the set of all possible solutions of the search space. At each iteration of the AAM search, we tune the blinking parameter between 0 and 1 with increments of 0.1. We calculate the corresponding fitness and we choose the set of C, T and T_{blink} parameters that give the minimum fitness to pass to the next iteration. The AAM search becomes the following.

- Initialize model parameters C_0 and T_0
- Set the number of iterations

While Number of iterations not reached **do**

- Calculate the initial residual error δg_0
- Predict displacements in C and T (δC and δT) according to equations 2.13 and 2.13 respectively
- for k = $[1, 0.5, 0.25]$ called the damping factor
  - Predict a new value of C and T: $C_{k_i} = C_0 - k\delta C$, $T_{k_i} = T_0 - k\delta T$
  - for $T_{blink} \in [0, 0.1, 0.2, \ldots, 1]$
    - Generate the model shape s_{model} and texture g_{model} using equations 2.2 and 2.6 respectively
    - Apply the pose vector on the shape coming from the model to obtain the shape on image s_{image}
    - Apply T_{blink}^{image} on the resulting s_{image} where

$$T_{blink}^{image} = T_{blink}^{model} \times scale_{imagetomodel} \tag{3.3}$$

$$T_{blink}^{model} = T_{blink} \times MeanEyeHeight_{model} \tag{3.4}$$

$$scale_{imagetomodel} = \frac{InterOcularDistanceInImage}{InterOcularDistanceInModel} \tag{3.5}$$

T_{blink}^{model} is the value of the blink parameter in the model space. It is approximated to be a fraction of the eye height of the mean shape: $MeanEyeheight_{model}$. T_{blink}^{image} being the value of the blink parameter in the image space, $scale_{imagetomodel}$

being the scale that maps a point in the image plane into the model plane. The $InterOcularDistanceInImage$ is calculated using the distance between the reality points corresponding to the eyes and the $InterOcularDistanceInModel$ is calculated by calculating the distance between the mean of the eyes points of the mean shape.
  - Map the pixels under s_{image} into the mean shape to form g_{image}
  - Calculate the corresponding residual δg_{k_i} using equation 2.11
  - Calculate the corresponding error E_{k_i} using equation 2.12
- Update C, T, and T_{Blink} by the one that gives the least error among C_{k_i} and T_{k_i}

In the following, we detail how we parameterize the actions of the eyeballs.

3.1.2 Multi-Object AAM

Continuing to generate movements of facial components without the necessity of including such movements in the learning base of AAM, we move to parameterizing gaze. In this section we show how we construct the AAM containing a gaze parameter at the level of one eye. This requires the fusion of 2 AAM, one for the iris texture and one for the surrounding skin (where a hole is put inside the eye). The following further details this approach.

3.1.2.1 Modeling

Most computer graphics methods model the eyeballs as separate objects from the facial skin (cf. figure 3.6). In these methods, the eyeball is located behind a $3D$ mesh that represents the facial skin and that has openings between the eyelids. In the facial analysis framework, particularly in the analysis-by-synthesis approaches, the eyeball is not modeled as a rotating $3D$ sphere located behind the skin surface. Instead, the visible region of the eyeball is a part of a continuous face mesh.

We inspire from the computer graphics community to parametrize the motions of the eyeballs to propose a multi-object representation: The basic idea of this representation is that the interior of the eye is considered as a separate object from that of the face which is another object. These two objects deform separately and are assigned each its own set of parameters. By permitting the eye object to slide under the skin surface object, we succeed at synthesizing any gaze direction and consequently at parameterizing the iris motion (cf. figure 3.7). An object in our case is either the eye skin or the iris texture. We thus choose to name our approach the Multi-Texture AAM.

To accomplish the decorrelation of the eye skin from the iris, the facial skin object should be modeled separately from the iris texture. Thus, we put holes inside the eyes in the place of the iris-sclera part. This permits to subtract the variability that the iris undergoes (scale, color and position) from the appearance parameters of the eye AAM.

Formally, equations (2.9) and (2.10) become

Figure 3.6: Illustration of modeling the eyeball as a sphere in computer graphics

$$s_{face} = \begin{pmatrix} s_{skin} \\ s_{iris} \end{pmatrix} = \begin{pmatrix} \bar{s}_{skin} \\ \bar{s}_{iris} \end{pmatrix} + \begin{pmatrix} V_s^{skin} C_{skin} \\ V_s^{iris} c_{iris} \end{pmatrix} \tag{3.6}$$

$$g_{face} = \begin{pmatrix} g_{skin} \\ g_{iris} \end{pmatrix} = \begin{pmatrix} \bar{g}_{skin} \\ \bar{g}_{iris} \end{pmatrix} + \begin{pmatrix} V_g^{skin} C_{skin} \\ V_g^{iris} c_{iris} \end{pmatrix} \tag{3.7}$$

where C_{iris} only encodes variations corresponding to the iris's appearance and shape. It has nothing to do with its scale and movements.

3.1.2.1.1 Local eye skin AAM

This model is built using 22 landmarks that describe the whole eye area, including the eyebrows and the texture surrounding the eye. Figure 3.11(a) is an illustration of the mean texture of the eye skin model showing the hole inside the right eye with the annotations to obtain this model.

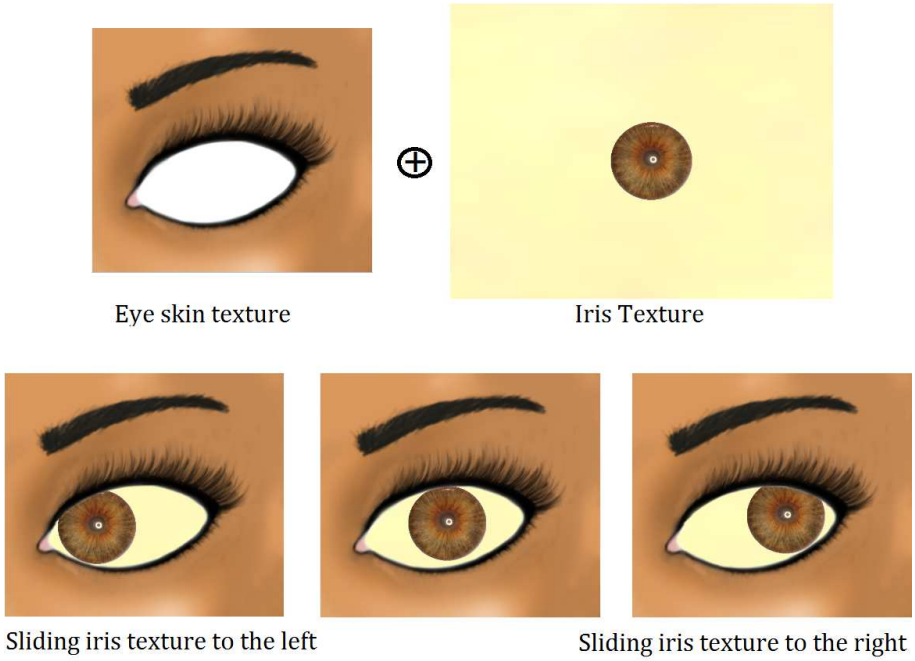


Figure 3.7: Multi-texture idea illustration. Moving the iris texture behind the eye skin surface creates different gaze directions.

3.1.2.1.2 Iris AAM

In order to build this AAM, we need an iris-sclera texture that is capable of sliding under the eye skin. We construct our iris training database starting from the iris images of [CCD00] and [DMTP04]. We reprocess these images to obtain the iris part. We then merge it with a white texture. We make the sclera texture by cropping a small area from the sclera in the original iris images and resizing it (cf. figure 3.8). Cropping from the sclera present in each image to reproduce the white texture results in different textures which are not totally white. This permits to learn different white information in the training phase of the iris AAM, thus making the model capable of coping with the variation of the sclera color from one person to another. The original iris images are of high resolution and they present unnecessary details in the iris texture. We resize these images and apply a circular averaging low pass filter to decrease the amount of information in the iris area.

For training these iris images we use a model of 13 landmarks of which 8 describe the shape of the iris in frontal view and 1 describes the approximative position of its center; to learn the white texture around the iris, 4 additional landmarks forming a rectangular shape around the iris are placed. Figure 3.11(b) is an illustration of the mean texture of the iris model with the corresponding annotations. These iris images are used in a $3D$ representation of the eyeball.

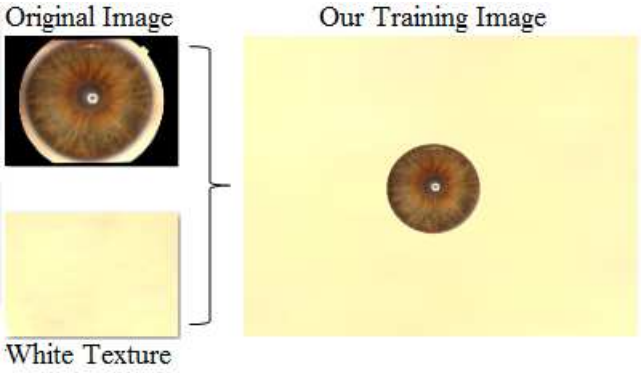


Figure 3.8: An example of a training iris image before and after processing (remark that the original image is of a very high resolution)

As we see from figure 3.7, when the iris texture slides to the extreme left or the extreme right using simple translation in $2D$, the appearance of the eye becomes unrealistic. In reality, the iris is a part of a spherical eyeball. As the eyeball rotates to extreme positions, the iris's appearance becomes elliptical rather than circular. Thus, modeling the iris in $2D$ is not sufficiently realistic and may cause problems in detection. This is why, we propose to model the iris as a part of a $3D$ eyeball (cf. figure 3.12).

Modeling the interior of the eye as a sphere and rotating it under the skin surface, any gaze direction can be synthesized. Consequently, the iris motion is parameterized and realistically modeled.

$$T^{iris} = \left[S^{iris}\theta_{hor}\theta_{ver}\right] \tag{3.8}$$

where S^{iris} is the scale of the iris, θ_{hor} is the horizontal rotation of the eyeball and θ_{ver} is its vertical rotation.

3.1.2.2 Searching: fusion of the eye skin and iris models

Fusion of the eye skin model and the iris one is done in the searching phase. First we find the optimal parameters for the eye skin (using the eye skin model) in a prior step. We then use the found parameters to reconstruct the image describing the eye skin. The iris model rotates under it with the pose vector T^{iris} describing the iris position with respect to eye.

To merge the eye skin object and iris models, we simply replace the hole in the skin model (figure 3.11(a)) with the pixels of the iris model (figure 3.11(b)). After replacement, a problem of discontinuity between the two models arises (cf. figure 3.9). As we see, the

resulting eye model seems unrealistic, especially at the borders of the eye skin model. In order to resolve this, we apply a circular averaging low pass filter of radius $R = 2$ on the skin and white parts while preserving the iris:

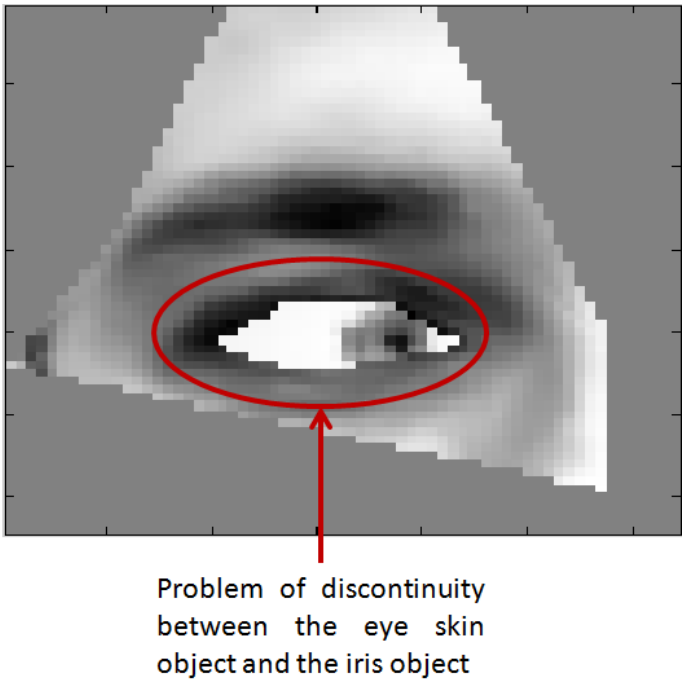


Figure 3.9: Discontinuity between the eye skin object and the iris object when merging them

$$h(x,y) = \begin{cases} 0 \text{ if } \sqrt{(x^2+y^2)} > R \\ \frac{1}{\pi r^2} \text{ if } \sqrt{(x^2+y^2)} \leq R \end{cases}$$

It smooths the discontinuity between the eyelid and the iris, and also reproduces the shadow effect of the eyelid on the iris. We remark that the filter is not applied on the iris since it is essential to preserve a good image quality of the iris in order to guarantee the localization. This is done using the mask shown in figure 3.11(d). The filter is applied on all the pixels of the white area of the mask. We remark that some pixels of the perimeter of the iris are non-intentionally affected by the application of the filter using the mask (cf. figure 3.10). This is because the landmarks of the iris are not abundant (8 landmarks). Thus, it causes the inclusion of some pixels of the perimeter of the iris in the region of application of the mask. However, this does not affect the detection since it only concerns few pixels. This results in the final model describing the eye region (figure 3.11(e)).

The following sums up the whole eye AAM fitting process using the $3D$ representation.

Algorithm

Steps for fitting the $3D$ MT-AAM for one eye:

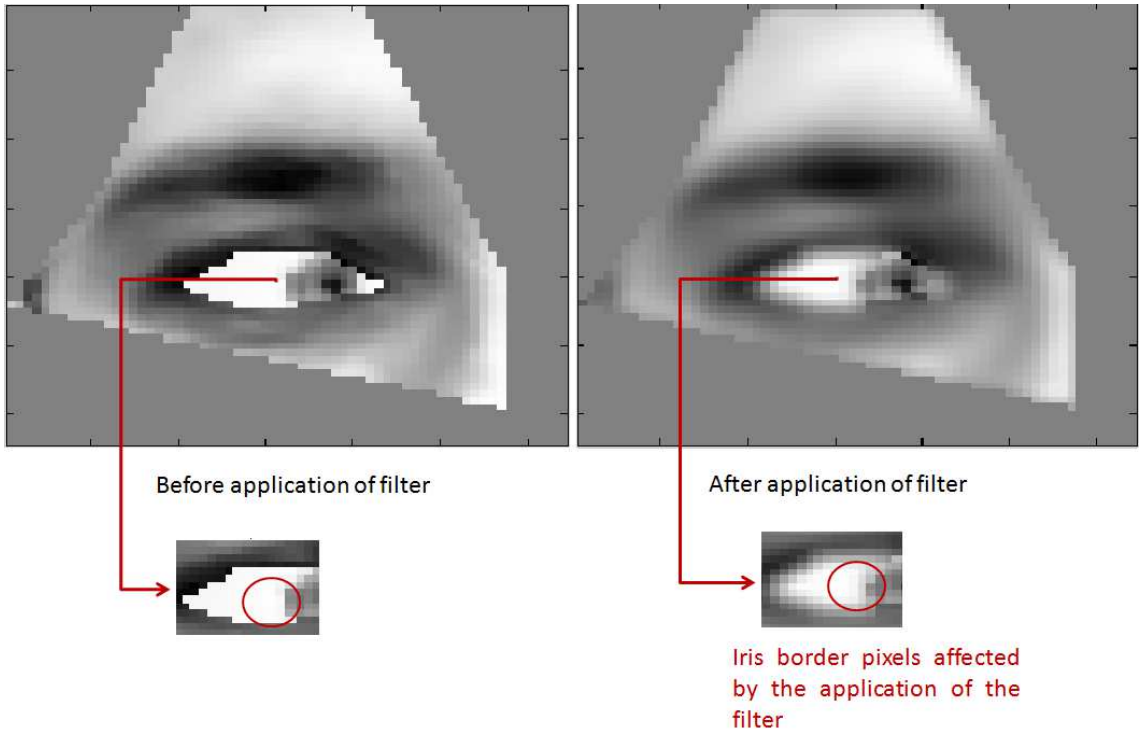


Figure 3.10: Iris border pixels affected by the application of the filter

1. Localize the eye using the eye skin model.
2. From the webcam image, extract the texture of the eye (g_i) (figure 3.11(f)).
3. Using the optimal parameters found by the eye skin model, synthesize the eye skin (g_m^{eye}) (figure 3.11(a)).
4. Until the stop condition (number of iterations reached) do:
   (a) Create the model texture of the iris (g_m^{iris}) based on the pose and the appearance parameters of the iris model (figure 3.11(b)).
      i. Project the $2D$ shape of the iris on the sphere (figure 3.12(i))
      ii. Rotate the iris and the sphere in $3D$ (figure 3.12(ii))
      iii. Project the $3D$ iris on $2D$ (figure 3.12(ii))
      iv. Map the iris texture on the rotated shape (cf. figure 3.11(b))
   (b) Merge the two textures g_m^{eye} and g_m^{iris} to obtain the texture g_m (figure 3.11(c)).
   (c) Apply a selected low pass filter to get the final eye region model g_m (figure 3.11(e))
   (d) Evaluate the error E: $E = g_i - g_m$ in the interior of the eye region (figures 3.11(g) and 3.11(h))
   (e) Tune the pose and appearance of the iris model.

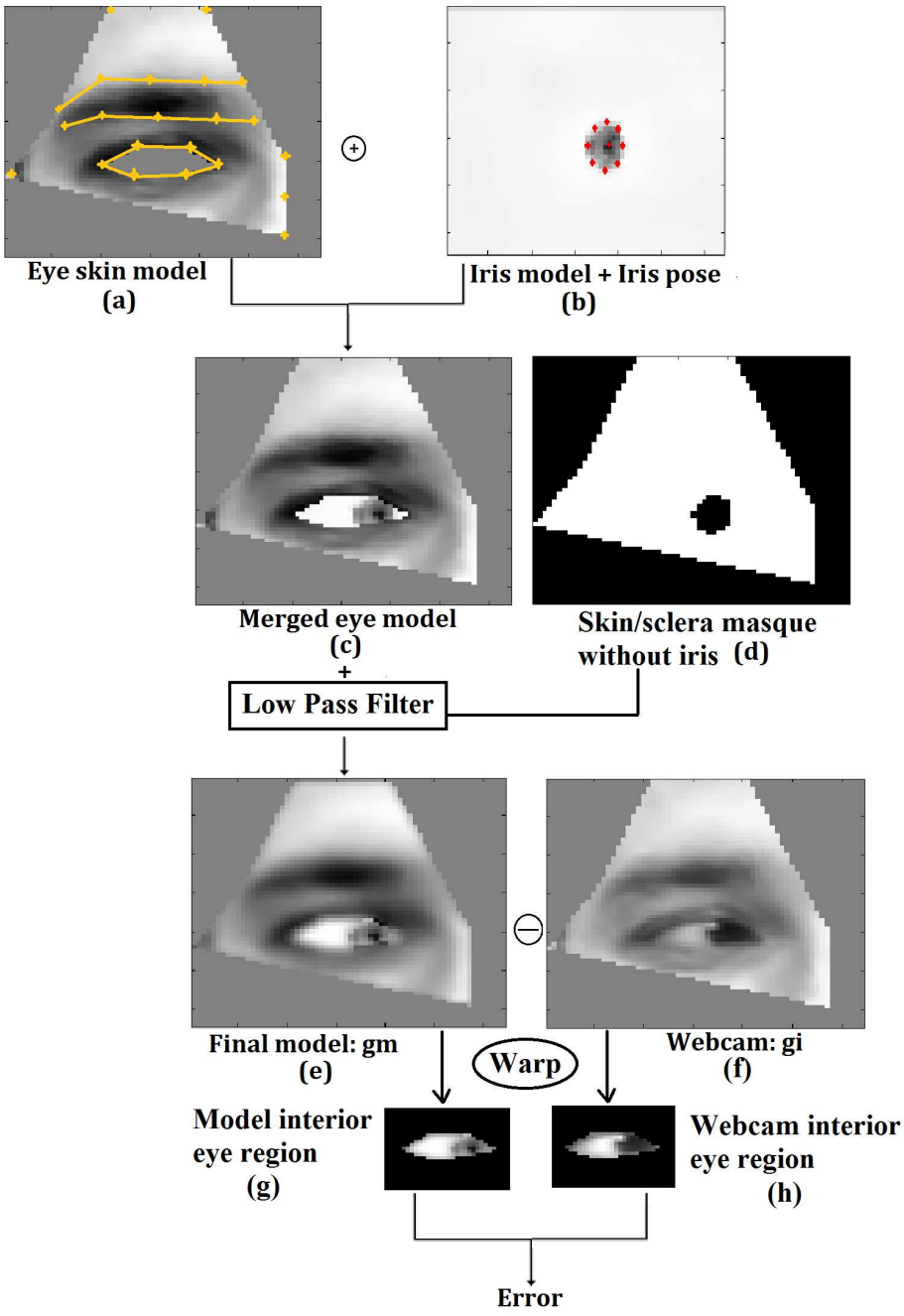


Figure 3.11: Error Calculation at one iteration. At each iteration, the eye skin model is merged with the iris one to obtain the final eye model which is compared to the real eye to get the error

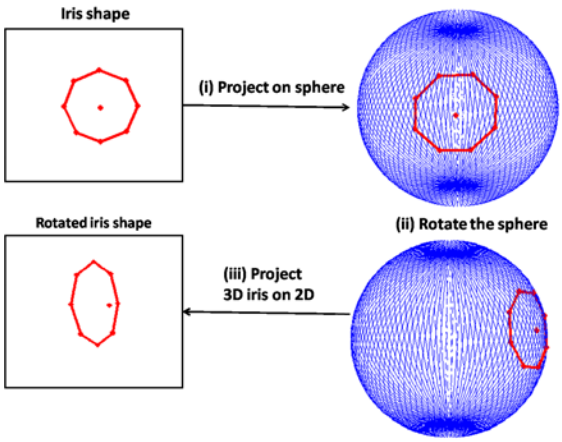


Figure 3.12: Illustration of modeling the iris as a part of a sphere. See how the iris appearance becomes elliptical in appearance for extreme gaze directions

Note that the evaluation of the error is computed in the interior of the eyes and not over the whole eye region. This is because the eyes skin texture is the optimal one, so it is unnecessary to include the skin texture in the error calculation. It will only add noise to the latter.

Eyeball diameter – The average diameter of the iris in the human eye is around $12mm$ and the average eyeball diameter is around $26mm$. Thus, we fix the ratio between the iris diameter and that of the eyeball to 0.45. We calculate the iris diameter from the mean shape of the iris model and deduce the eyeball's diameter using the fixed ratio.

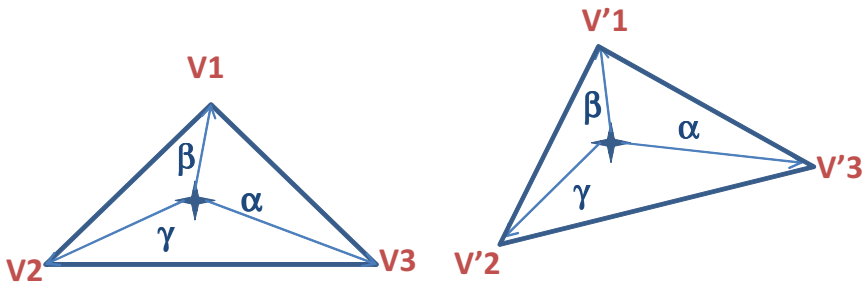


Figure 3.13: Barycentric coordinates

Back projection of iris shape When the optimal iris shape is found in the model frame, it should be back-projected to the real image. Actually the optimization of the iris parameter takes place in the model frame where the eye texture is warped from the web-cam image frame to the mean model of the eye. The iris position is then optimized

with respect to the midpoint of the mean eye shape. In order to retrieve the iris position in the original image using the iris parameters found in the model frame, we use barycentric coordinates. This is the same approach used in warping from one image to another. Let S_{iris}^{model} be the optimal iris shape found in the model frame and S_{iris}^{image} the iris shape in the real image. To find this latter:

- Perform Delaunay triangulation on the S_{eye}^{model}
- For every point $s_i^{model} \in S_{iris}^{model}$
  - Determine the triangle that s_i^{model} belongs to among the triangles of the Delaunay triangulation:
    - Transform the coordinates of $s_i^{model}\,(x, y)$ into barycentric coordinates of each of the triangles of the Delaunay triangulation. Let α_t, β_t, γ_t be the barycentric coordinates of s_i^{model} w.r.t each triangle having vertices r_i of cartesian coordinates (x_i, y_i), $i \in 1 \ldots 3$, then s_i^{model} belongs to triangle t if $0 < \alpha_t, \beta_t, \gamma_t < 1$.
    - Find the position in the original image: $s_{iris}^{image} = \alpha_t r'_1 + \beta_t r'_2 + \gamma_t r'_3$ where $r'i$ are the vertices of the triangle t' that s_{iris}^{image} belongs to and that corresponds to t

3.1.3 Multi-Objective modeling: general idea

We have seen in the previous section how dealing with the eyeball as a different object from that of the eye skin permits the coding of gaze in a separate vector of parameters than those of the AAM appearance parameters. In this section, we show another aspect of multi-object modeling. This aspect concerns partitioning the facial skin itself to different objects and linking these in a multi-objective optimization framework.

As a matter of fact, one of the challenges of AAMs is to be able to deal with changes of appearance that the face would be submitted to. These appearance changes might be due to occlusions such as head pose. As the face is changing its pose, some features of the face might be partially or totally occluded.

A multi-object representation of the face helps promoting one or several objects over the others in the presence of head pose. This works as follows: the face is partitioned into several objects, and each object has its own residual error. According to certain factors (such as occlusion of a feature of the face), one or several features are favored over the others.

If $E_1, E_2, ...Ei$ are the residual errors of each "object". Optimizing these errors leads to a multiobjective optimization. Each error is assigned a weight and the resulting errors are added to form one final error. The weights serve at favoring one object over the others. In this way, occluded parts of the face can be penalized less and thus they contribute less to the final error.

$$E_{final} = \alpha_1 E_1 + \alpha_2 E_2 + \ldots \alpha_i E_i \tag{3.9}$$

where i is the number of defined objects of the face. αi is the weight assigned to each object.The number of different objects and the weights can be defined depending on the nature of the application and its requirements. We will show in section 3.1.3.1 how we define these in the context of a gaze detection application.

We note that this approach can be related to hierarchical models (cf. section 2.3.1.4) where each region of the face is assigned a separate model. However, non of these methods relate the different regions to each other through a multiobjective optimization.

3.1.3.1 Integration in the gaze detection system

Let us consider that a subject is in front of the screen where a webcam is installed (first block of figure 3.14). Depending on the face orientation, the left and right eyes are unevenly represented in the webcam image. In other words, the face orientation in this case causes partial or complete occlusion of one eye with respect to the other. As we are concentrating on gaze, then there is no need to integrate the lower part of the face in the model. Thus, we partition the upper region of the face into two: the left eye region and the right eye region. The gaze is analyzed using a multi-objective optimization with local models containing gaze parameter for each eye (one Multi-Texture AAM (MT-AAM) for each eye): The contribution of each eye to the final gaze direction is weighted depending on the detected face orientation.

Figure 3.14 depicts the steps of our global system. We distinguish between three main zones depending on head orientation.

(A) The subject could have a head pose such that both of his eyes appear clearly on the screen;

(B) The subject shows a large head rotation to the left such that the right eye appears the most in the camera;

(C) The subject shows a big head pose to the right such that the left eye appears more;

The algorithm works as follows: The first step is the detection of the head pose for which a $2.5D$ global AAM model ([SALGS07a]) is applied. If the head pose corresponds to the Zone A, then both eyes will be integrated in the detection of the gaze using a weighting function. If it is in the Zone B, an MT-AAM will be applied only on the right eye. If it is in the Zone C, then MT-AAM will be applied only on the left one.

3.1.3.2 Multi-objective optimization

Since normally the two eyes have highly correlated appearance and motion, we use only one pose vector and one appearance vector to describe the pose and appearance of both irises. Technically, it should be sufficient to analyze the iris of one eye to obtain its position and appearance in both eyes. Yet, the information from both eyes can lead to a more robust system especially when the person commits large head movements around

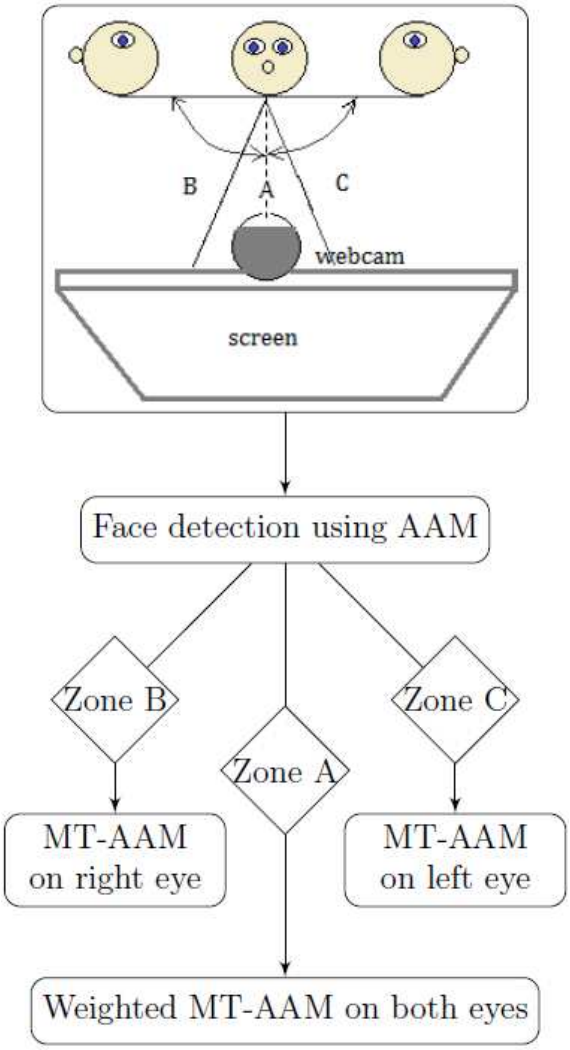


Figure 3.14: Global system overview; Zone A means that the face rotation is sufficiently small such that both eyes appear in the camera. Zones B and C signify that the right or the left eye appear more in the camera respectively

the vertical axis where one of the eyes can be partially or completely occluded. This is solved by the multi-objective AAM framework presented in section 3.1.3.

The idea is that we deal with the eyes as if they were two separate images acquired at the same time: MT-AAM (see section 3.1.2) is applied simultaneously on both eyes and at each iteration, and the resulting errors are summed while multiplying each by a weighting factor that is a function of the head pose.

In this system, a single iris model is merged simultaneously with the left and the right

eye skin models. Then the resulting models are overlaid on both the right and the left eyes from the camera to get the left and right errors. These are weighted according to the head orientation and summed to get one global error. This error becomes:

$$E = \alpha E^{left} + \beta E^{right} \tag{3.10}$$

where E^{left} and E^{right} are the errors corresponding to the left and right eyes respectively. α and β are the weighting factors. They are functions of the head rotation around the z-axis (R_{yaw}), evaluated just after the face detection, and they both follow a double logistic law:

$$\alpha(R_{yaw}) = \begin{cases} 0.5 \text{ if } -d \leq R_{yaw} \leq d \\ 0 \text{ if } -90^\circ < R_{yaw} \leq -22^\circ \\ 1 \text{ if } 22^\circ \leq R_{yaw} < 90^\circ \\ 0.5(1 + l(1 - \exp \frac{-(R_{yaw} - ld)^2}{\sigma^2})) \text{ else} \end{cases}$$

$$\beta(R_{yaw}) = 1 - \alpha \tag{3.11}$$

where $l = \text{sign}(R_{yaw})$, σ is the steepness factor and d is the band such that the two functions α and β are equal to 0.5. d is chosen to be 7° such that for this value we consider that the orientation of the head is negligible and that both eyes contribute equally. A head rotation of 22° is considered to be big enough to make one of the eyes appear more than the other in the image, and thus, this eye is exclusively taken into consideration in the detection of the iris. σ is found empirically. In this way, the face orientation is taken into

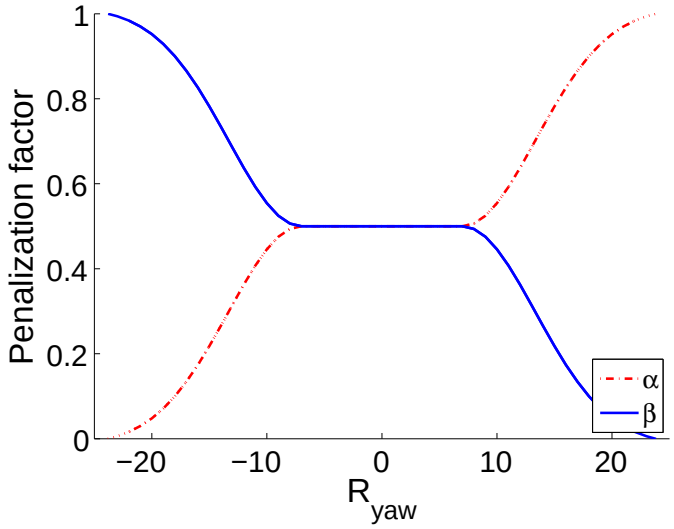


Figure 3.15: Double logistic function, d = 7. R_{yaw} is in degrees.

account by the relevant information from both eyes.

To minimize the error in equation (3.1.3.2), we have tried several optimizations: Gradient Descent (GD), Simplex, and Genetic Algorithm (GA). After comparison of these (cf. sectionsec:optimizationcomp), we have chosen the genetic algorithm to optimize the proposed gaze pose vector.

Genetic Algorithm (GA) [Rec71] is a population-based iterative search heuristic that aims at finding the set of parameters that optimizes a certain cost function. It is inspired from the process of natural evolution. A set of candidate solutions evolves towards a better set until arriving at the best one.

The iris pose and appearance form the genes of the GA. They are combined into a same vector to form a chromosome (cf. figure 3.16).The group of chromosomes form a population. At each iteration, a population of solutions is formed. Their corresponding fitness is computed. According to the latter, some of these chromosomes are set to be the parents of the population of the next iteration.

Figure 3.16: A chromosome of the genetic algorithm; The parameters inside are those of the iris appearance and pose

Briefly, the options of the GA that we found to be the best suited to the optimization of the iris parameters are the following:

Initialization: The initial population is generated randomly between the upper and the lower limits of the parameters with a uniform distribution. This allows to span all the search space.

Selection: This is the act of choosing the parents for the production of the new generation. We use tournament selection of which a number of chromosomes are randomly selected from the population, their fitness computed, and the ones having the least error are selected to be included in the next generation population.

Reproduction: This is done by two-point crossover and mutation with a proportion of 0.8 for crossover 0.2 for the other. Crossover combines existing chromosomes (2 parent chromosomes are combined to a child chromosome). Mutation performs random small changes to a chromosome. We use Gaussian mutation.

3.2 Tests and Results

As our concentration was on the eye region, our experiments concern it as well. In this section, tests pertaining blinking detection are performed (section 3.2.1). In addition, we present results concerning the gaze detection system presented in this chapter (section 3.2.2). Since we conduct many experiments, we present in table 3.1 a summary of the training and testing databases used in each experiment.

Table 3.1: Summary of the training and testing images used in the different experiments

	Section	Experiment	Training		Testing	
			Database	No. of samples	Database	No. of samples
Blink	3.2.1.1		Database 1	9	Database 1	81
	3.2.1.2		Database 1	9	Database 1	81
			Database 2	5	Database 2	65
	3.2.1.3		Bosphorous	88	PG	68
Gaze		Iris model	[DMTP04]	23		
	3.2.2.1	Eyes skin model	Bosphorous	104 neutral	PG	100
					UImHPG	185
	3.2.2.2	Eyes skin model	Bosphorous	104 neutral	PG	172
	3.2.2.3	Eyes skin model	Bosphorous	104 neutral	PG	98
	3.2.2.4	Eyes skin model	PG	10	PG	100
					UImHPG	185
		DEAAM	PG	50	PG	100
					UImHPG	185
	3.2.2.5	Eyes skin model	Bosphorous	104 neutral	UImHPG	129

3.2.1 Blink detection

The purpose of this section is to test the capability of the blinking parameter that we add to the AAM to detect the different states of the eye, both in person-specific and generalization cases.

Concerning the person-specific case, learning and testing are done using two small databases that we make in our laboratory for the task of testing our model. Database 1 is constituted of 10 subjects filmed at 60 fps using the Hercules webcam [Her].

Database 2 is constituted of 5 subjects filmed at 120 fps using an InfraRed Camera called OptiTrack [Opt].

As for the test in generalization, we use the Bosphorous database [SAD+08a] for learning and the Pose Gaze database [ASKK09] for testing.

The objective of these tests is to prove that indeed adding a blinking parameter to the AAM surpasses the necessity of including subjects performing blinks in the learning database in order to track such variation.

In the following, first we compare combinations of different optimizations and eye landmarks configurations for the blink model in section 3.2.1.1. This helps us choose the best combination for the blink model. Then we integrate the chosen model in a global face model in section 3.2.1.2. Finally, in section 3.2.1.3 we test the blinking model in generalization.

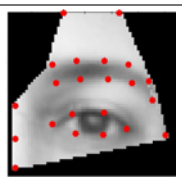
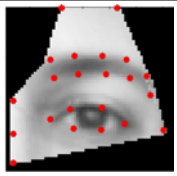
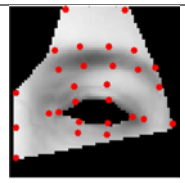
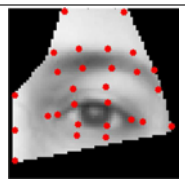
Models		RB1	RB2	RB3	RB4
Opt.	$C\&T$	GA	GA	Regression	Regression
	T_{Blink}	GA	ES	ES	ES
					

Table 3.2: Summary of the different eye blink models with different optimizations and configurations. $C\&T$ are the appearance and pose parameters of the eye region. T_{Blink} is the blinking parameter added to the AAM parameters. RB: Right Blink model, GA: Genetic Algorithm, ES: Exhaustive Search.

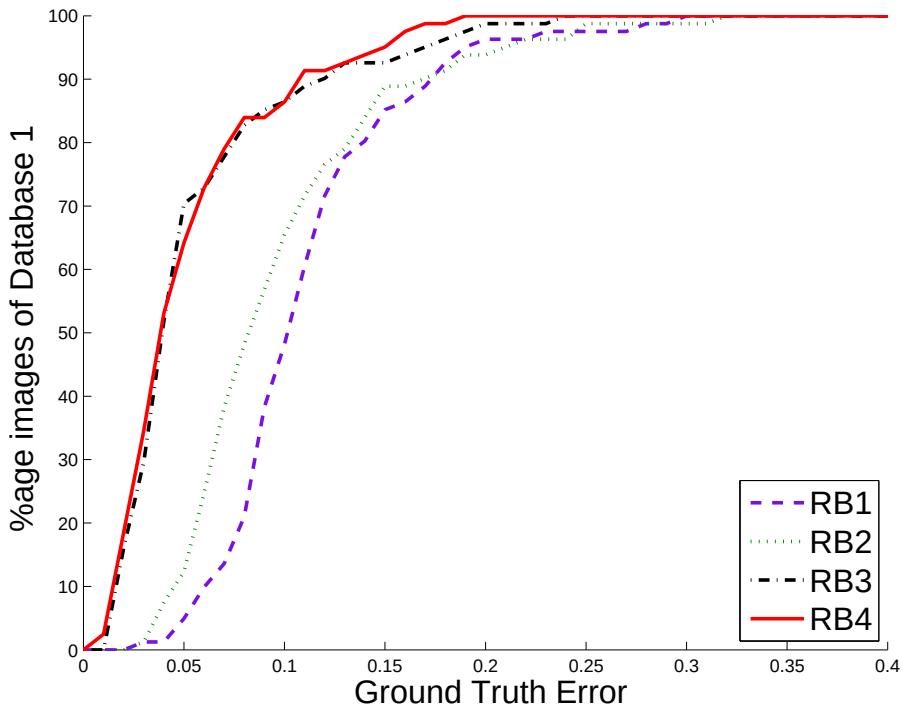


Figure 3.17: Comparison between the GTE of different eye models

3.2.1.1 Comparison between different one eye models

In order to drive conclusions about what optimization method to choose for the blink model and what points to put around the eyes, we compare the performance of several eye

models. These models are local models of the right eye. We choose to do the comparison using a local model and not a global one so that the AAM would be concentrated on the eye area. The comparison would then be more confident. The results are then generalized to a global model of the face. The different models that we compare are summarized in table 3.2. Model $RB1$ is a model that uses GA algorithm optimization for both the eye appearance and pose ($C\&T$) and the blink parameter T_{blink}. Model $RB2$ uses GA for the $C\&T$ but exhaustive search for T_{blink}. Model $RB3$ uses the classical regression matrix optimization for $C\&T$ together with exhaustive search for T_{blink}. Finally, model $RB4$ is the same as model $RB3$ concerning optimization, however it differs from the latter in that there is no hole inside the eye. The reason for this last comparison is to see if there is an effect of putting a hole inside the eye for the blinking parameter or not.

As we have indicated in the beginning of the chapter, the objective of adding a blinking parameter is to restrict the AAM database. The AAM will then be able to follow the subject's eyelids during blinking without the necessity of including such variations in the learning database. In addition, we will have the information of blinking encoded in this parameter, permitting the direct use of this information in different applications. Thus, as a starter, and to show the capability of the proposed modeling to follow the eyelids, we build an AAM using only subjects opening their eyes and we test on the same subjects while they are performing blinking. The subjects used are those of Database 1 already mentioned at the beginning of section 3.2.1. So, the model is built using the 10 open-eyed images of this database, and the test is done on the rest.

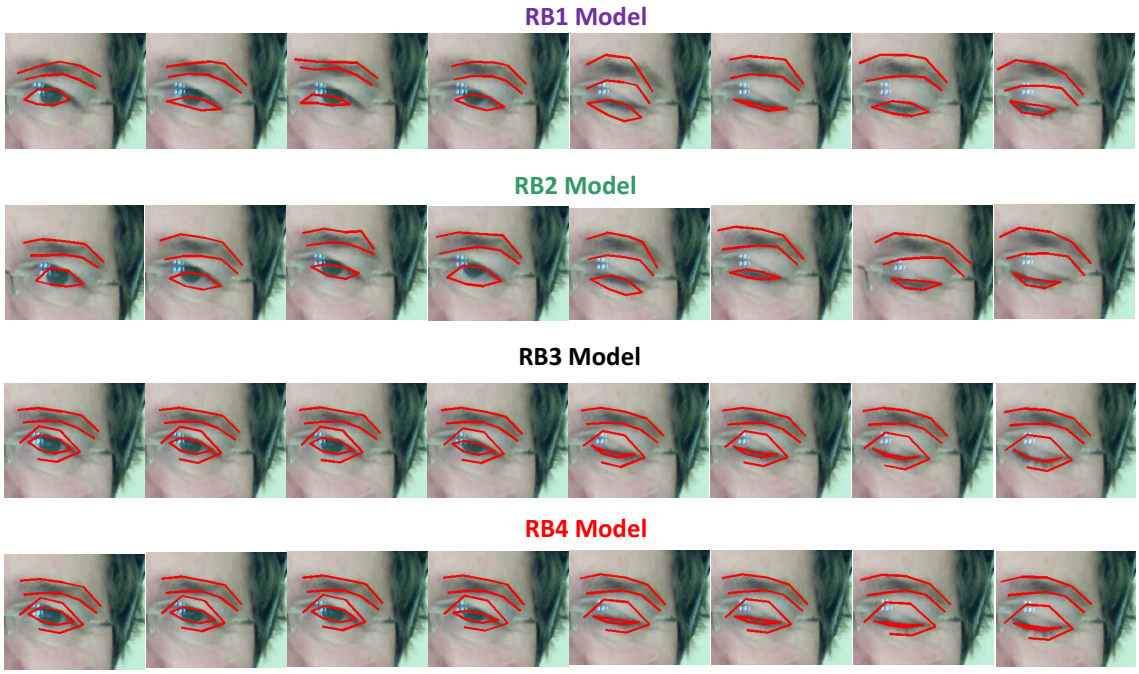
We compare the Ground Truth Error (GTE) of the eyelid. The GTE_{eyelid} is defined as follows:

$$GTE_{eyelid} = \frac{mean(\{d_i\}_{i=1:6})}{d_{eyes}} \tag{3.12}$$

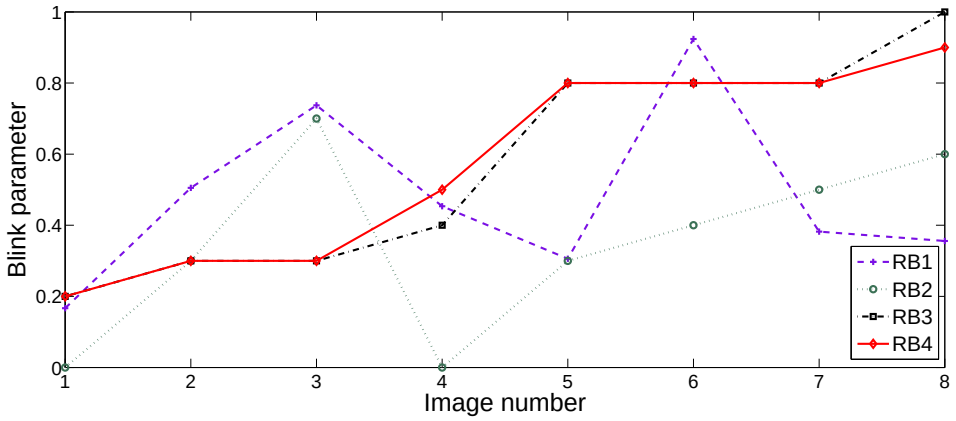
where $\{d_i\}$ are the distances between the ground truth and each of the found points by the eye skin model. d_{eyes} is the distance between the eyes when the person is looking in front of him.

Figure 3.17 presents the GTE_{eyelid} comparing the 4 aforementioned models. The curves show that for an error less than or equal to 10%, RB1 was to able localize the eyelids in 48.15% of the images in the database with 65.43 for $RB2$, and 86.42 for both $RB3$ and $RB4$. This signifies the outperformance of models $RB3$ and $RB4$ over the others. However, having equal detection percentage for RB3 and RB4 does not mean they have the same performance, since for 11%, RB3 has a good detection of 88.89% of the total number of images versus 91.36% for RB4. This means than a model without a hole performs slightly better that a model with a hole in some cases when there is blinking. The reason is that for intermediate eyelids movements (i.e when the blink is its middle state), the information of the interior of the eye contributes in the calculation of the error leading the model to a better localization of the eyelids. As a matter of fact, for intermediate states of the eye, the corresponding error would be confusing for the model where these two states might give the same error for two differnt states. Furthermore, one should keep in mind that in the presence of gaze, a model with a hole was found to be better than that of a one

without a hole. This will be shown in section 3.2.2.1.1.



(a) Comparison between the different eye blink models



(b) Comparison between the blinking parameter corresponding to one sequence of the testing database using different eye blink models

Figure 3.18: Comparison between different eye models with a blinking parameter

On the other hand, figure 3.18a presents the eyelids tracking results of the eye of one sequence of the testing database while the subject closes his eyes. It is clear that models RB3 and RB4 succeed the most in following the subject's eyelids. Visually, these two

models outperform the other ones.

We also plot in 3.18b the blinking parameter for this sequence. As we see, all the models detect an open eye in the beginning of the sequence (models RB1 and RB2 detect a value of 0 of T_{Blink} and models RB3 and RB4 detect a value around 0.2 which means an open eye. However, the former models do not succeed at giving the significant value of the parameter throughout the entire sequence whereas for RB3 and RB4, the blinking parameter increases gradually to reach a high value (1 for RB3 and 0.9 for RB4) which indicates that the eye is closed. Detecting a linear evolution of the blink parameter permits a robust detection of the act of blinking. This is useful for applications that detect drowsy eyes while driving.

These results led us to adopt the optimization used in models RB3 and RB4 that is classical regression for the appearance and pose parameters of the eye with exhaustive search for the blinking parameter.

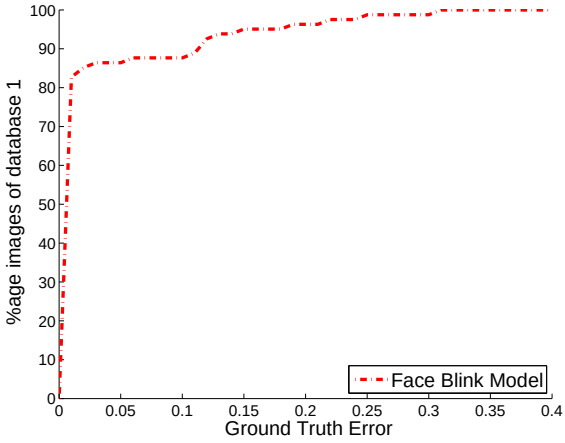
3.2.1.2 Integrating blink parameter into the whole face

Now that we have figured out the best combination of optimization for the blinking parameter and the face's appearance and pose, we integrate the blinking parameter into a global model of the face. The reason is that, first, we want to see the behavior of a global AAM with the addition a blinking parameter, and second, our final objective is one global model of the face that have motion parameters. Thus, we integrate T_{Blink} into a global AAM and we perform tests on our specific databases: Database 1 and Database 2. Figure 3.19 presents the GTE of the eyelids for these two databases. These curves show the capability of the model to follow the motion of the eyelids. As we see, the curves are close to the y_{axis} at the beginning which complies with the conclusions that we drew in the previous section. We compare the performance of the Face Blink global model with that of the local model RB4 (the model that gave the best result) for Database 1 in 3.19c. This figure shows that in this case the global model performs better than the local model which shows the interest of integrating the blink parameter in a global face.

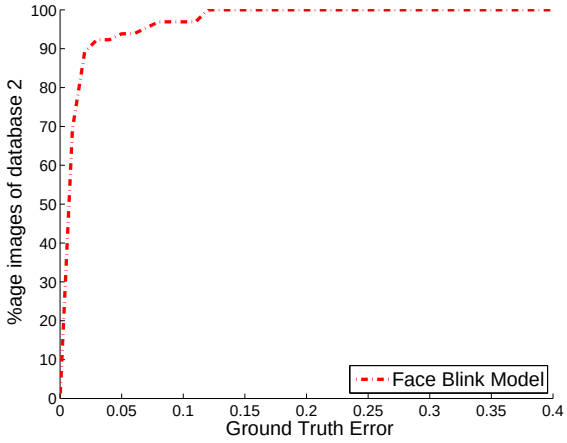
Figure 3.20 shows qualitative results on some of the test subjects. The efficiency of the added parameter is obvious for most of the subjects. However, the same figure shows one subject (the last one in the bottom) of which the model fails to localize the eyelids and follow the blink. The reason of this deficiency is in the subject himself. Actually, this subject has a very light appearance at the level of the eyebrows and eyes where the eyebrows are not very thick. Consequently, it is hard for the model to accurately align the eyes and eyebrows.

3.2.1.3 Test in generalization

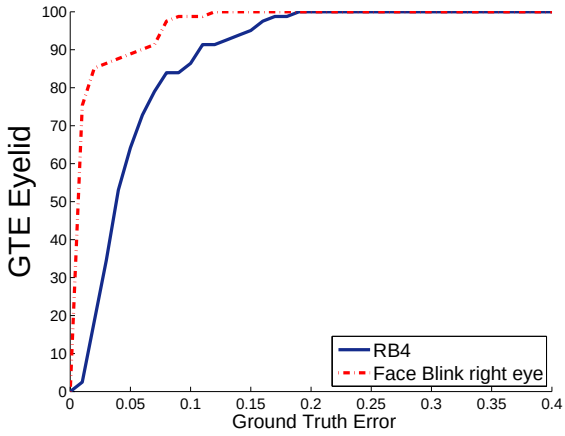
No matter how efficient the model is in the person-specific case, it is always challenging to be able to generalize to new data. To test the efficiency of the model in generalization, we constitute a training database using 88 images of the Bosphorous database [SAD+08a].



(a) GTE eyelids of the Face Blink model together with that of RB4 for database1



(b) GTE eyelids for database 2



(c) Comparison between RB4 and Face Blink model for database 1

Figure 3.19: GTE eyelids

Concerning the testing images, the images of the PG database [ASKK09] of which the subjects blink their eyes were chosen. Figure 3.21 shows the GTE of the eyelids detection for this database for both Face Blink models: with and without a hole. Actually since in section 3.2.1.1 it was not visually clear where a model with a hole performs better than a model without a one, we would like to compare this visual performance in generalization. The curves of figure 3.21 show that also here, a model without a hole is better suited for tracking the eyelids (79.41% vs. 76.47% at 10%) when there is blinking which conforms the curves in the person-dependent case.

Figure 3.22 shows some qualitative results on some images of the PG database. We can

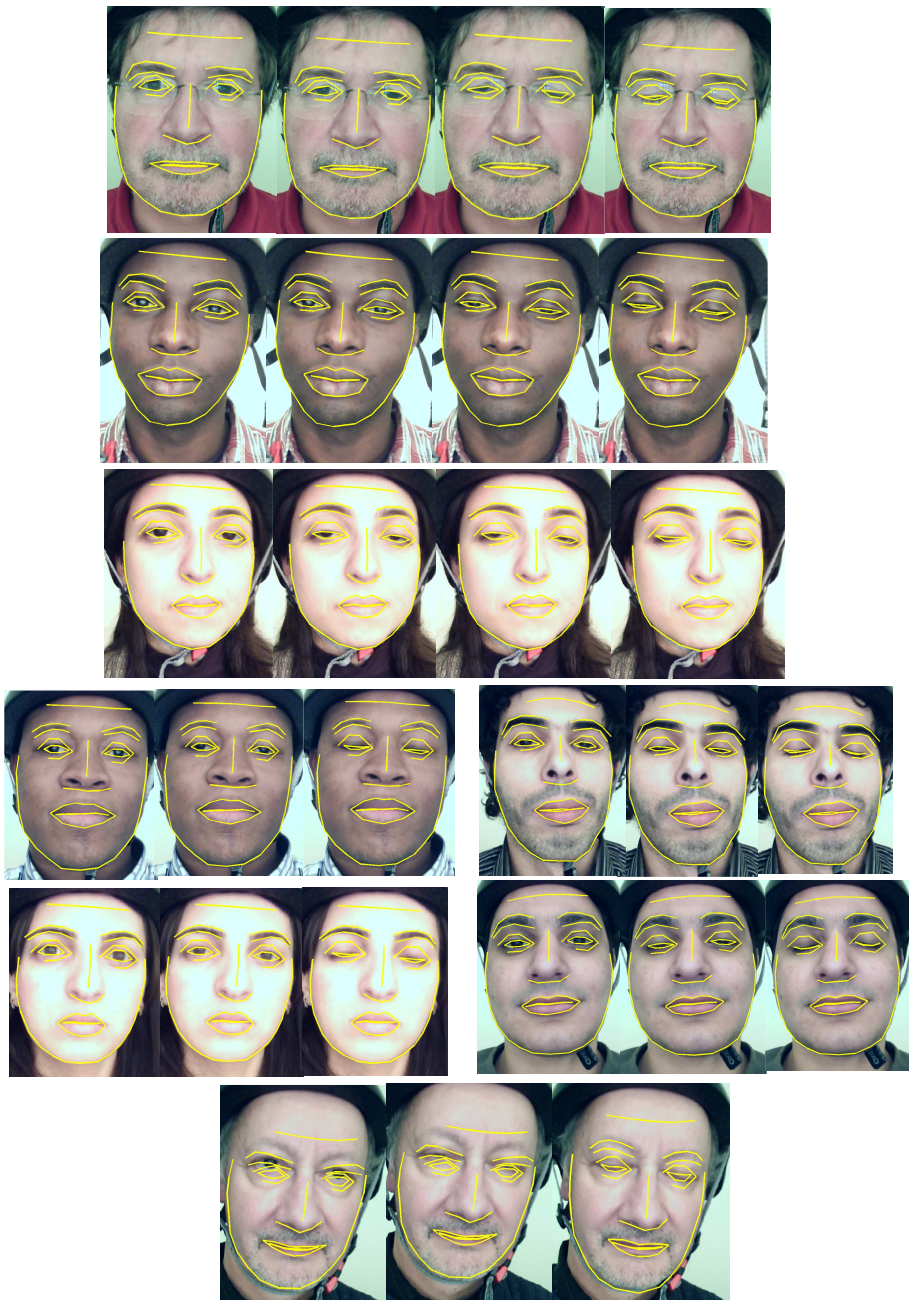


Figure 3.20: Results of the Face blink model on Database 1

see from these examples that integrating the blink parameter is efficient in following the eyelids. We remark that even when the model does not succeed at estimating the head pose, the eyelids motion is well followed.

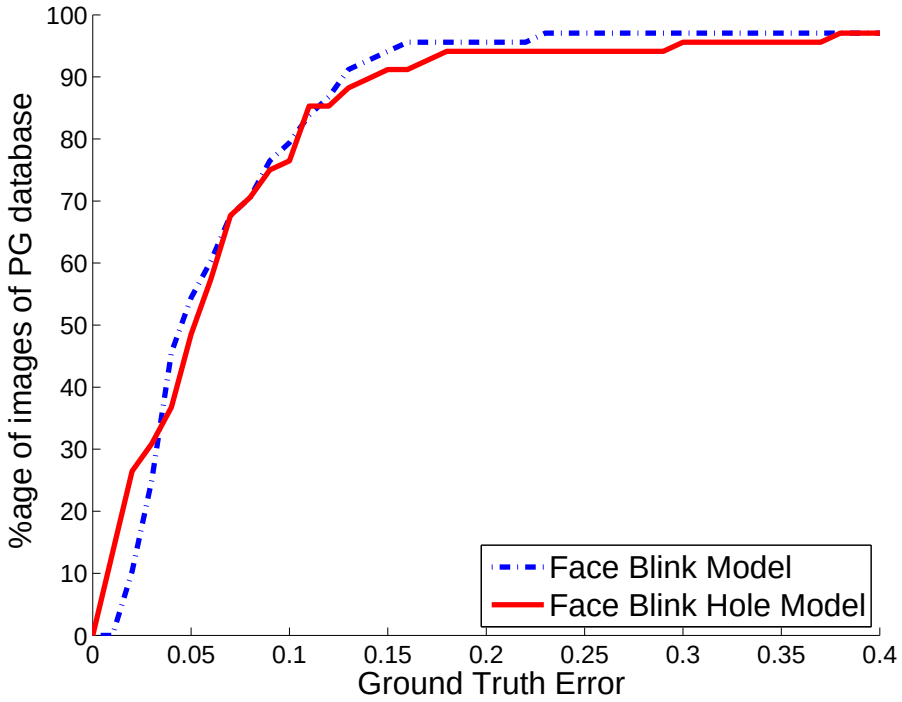


Figure 3.21: Ground Truth Error of the Face Blink model, testing on the PG database

Moreover, figure 3.23 presents examples where the model with hole is more efficient than a model without a hole. In these examples, the subjects gaze is to the extreme left. Since the model without a hole takes into consideration the interior of the eye and the training database was made on subjects looking in front of them, testing on a gazing subject is misleading for this model causing it to fall into a local minimum. On the other hand, the model with a hole does not take into consideration the interior of the eye, accordingly, it is more robust in such cases.

The question remains: As the final objective is a model that is capable of detecting blink and gaze at a time, what is the most suited approach?

3.2.2 Gaze detection

Concerning the gaze detection system presented in this thesis, we conduct five experiments: one to check the accuracy of the eye skin model and the dependence of the proposed model on the eye detection method (section 3.2.2.1), one to compare different optimizations of the proposed Multi-Texture AAM (section 3.2.2.2), one to test the Multi-Objective AAM versus Single-Objective (section 3.2.2.3), one to compare the $2D$ Multi-Texture AAM, $3D$ Multi-Texture AAM and a classical AAM for iris detection (section 3.2.2.4),

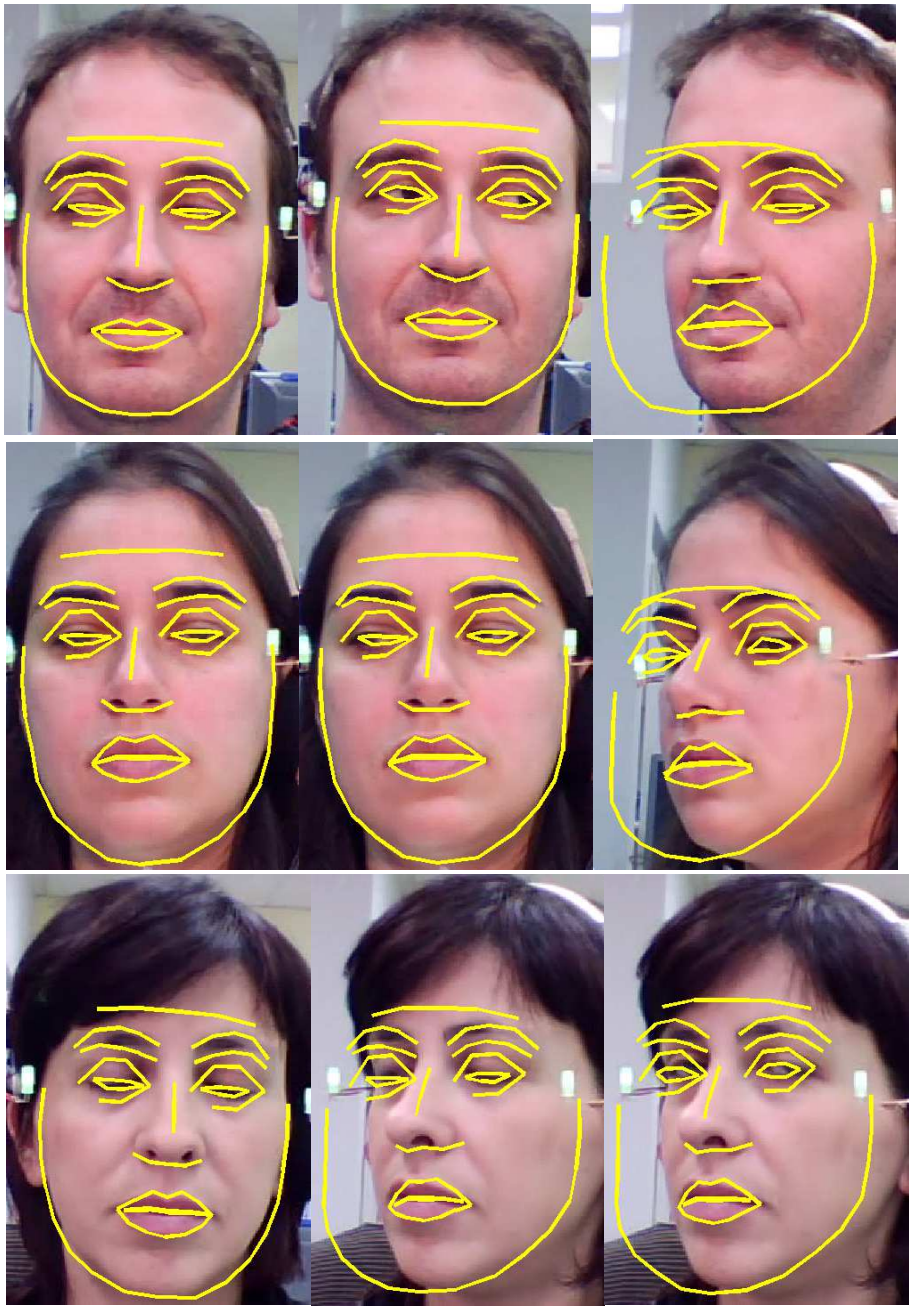


Figure 3.22: Results of the Face Blink model in generalization for some images of the PG database.

and finally, one to compare the 3D-AAM to a state-of-the-art method (section 3.2.2.5).

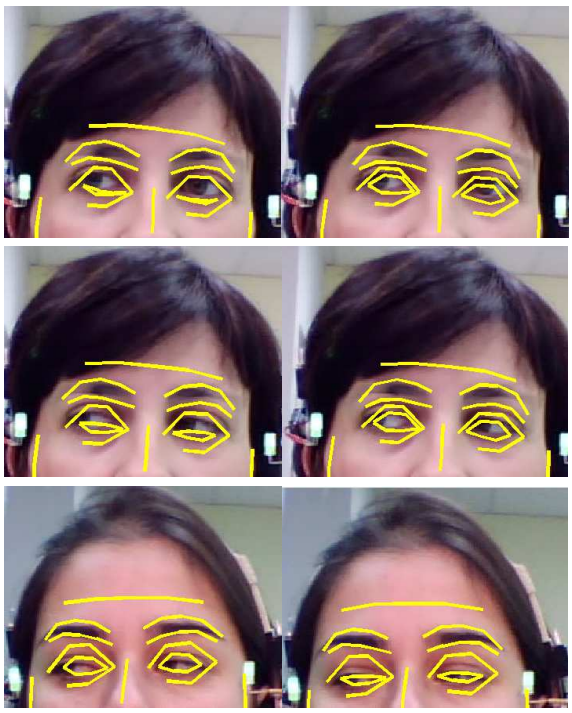


Figure 3.23: Visual result showing comparison between the Face blink model with and without a hole. The image to the left corresponds to the model without a hole while the image to the right corresponds to a model with hole.

All of these tests are done in generalization. The third test comparing the MT-AAM to the classical AAM is also done by training and testing on the same database.

Tests are conducted on the Pose Gaze (PG) database [ASKK09] and the UIm Head Pose and Gaze (UImHPG) database [WLSN07]. The first database was recorded using a simple monocular webcam. It contains 20 color video sequences of 10 subjects committing combinations of head pose and gaze. The second was recorded using a digital camera. It contains 20 persons with 9 vertical and horizontal gaze directions for each head pose of the person (range $0°$ to $90°$ in steps of 10 for the yaw angle and $0°$,$30°$,$60°$ azimuth and $-20°$ and $20°$ elevation for the pitch angle).

3.2.2.1 Accuracy of the eye skin model

This section discusses two issues. The first one concerns studying the effect of using an eye skin model (holes put in place of sclera-iris region): a comparison in generalization with a classical local eye model where the appearance of the interior of the eye is learned with that of the face skin is done. The second one studies the dependence of the MT-AAM on the localization of the eyelids. The first issue serves at showing that putting

holes instead of the sclera-iris increases the accuracy of the eyelids localization since it removes perturbations in the appearance due to gaze. On the other hand. The second has as objective to show how dependent the MT-AAM is of the eye localization method.

3.2.2.1.1 Eyelids localization: model with holes vs. model without holes

In order to compare the performance of the eye skin model where holes are put inside the eye to that where the interior of the eyes is kept, we plot the ground truth error of the eyelids versus the percentage of images in the database.

For this test, both models were trained on 104 neutral images of the Bosphorous database.

Figure 3.24 shows the GTE_{eyelid} of both methods for both right and left eyes. Tests were done on 100 images of the PG database and 185 images of the UImHPG database. The figure shows that for the four cases (left and right eyes of both databases) we have a higher GTE curve in the case of an eyelid model with a hole inside the eye.

Figures 3.25a and 3.25b show qualitative results of both models on some images of the PG and UImHPG databases respectively. As we see from the figure, the eye skin model finds the good localization of the eyelid while the model without a hole does not. The reason is that the information inside the eye (color of the iris and the different iris locations) influence the localization of the points of the eyelids when the interior of the eye is learned with the model. We can see from these results how the model always follows the position of the iris and so it diverges. By deleting this information, we have succeeded to delete its disturbance and we are able to better localize the eyelids.

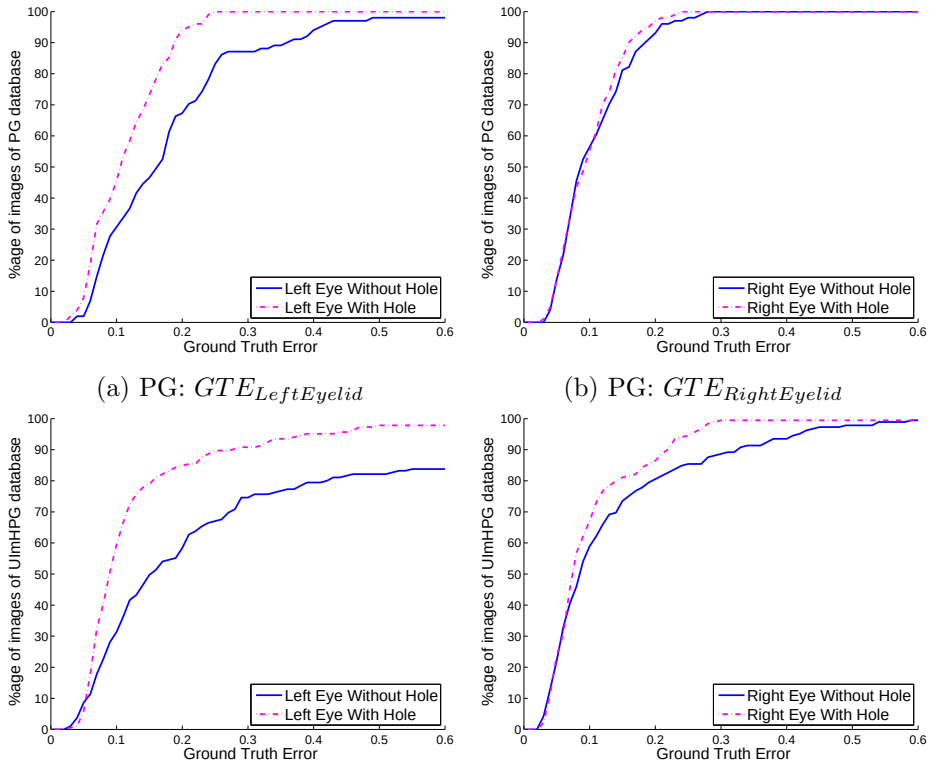
3.2.2.1.2 Dependency of the MT-AAM on the eyelids localization

Let $GTE_{2eyelids}$ be the Ground Truth Error of the two eyelids calculated from the GTE_{eyelid} of the right and the left eyes. Actually, according to the eye that was used in the detection of the iris, the corresponding GTE_{eyelid} is taken into account.

$$GTE_{2eyelids} = \begin{cases} mean(GTE_{lefteyelid}, GTE_{righteyelid}) & \text{if } -d \leq R_{yaw} \leq d \\ GTE_{righteyelid} & \text{if } -90 < R_{yaw} \leq -22 \\ GTE_{lefteyelid} & \text{if } 22 \leq R_{yaw} < 90 \\ \alpha GTE_{lefteyelid} + \beta GTE_{righteyelid} & \text{else} \end{cases}$$

R_{yaw} is the horizontal head pose, α and β are the weights calculated using the double logistic function, and $d = 7°$ is the band such that α and β are equal to 0.5.

Let the GTE_{iris} be the mean of the distance (Euclidean distance) between ground truth (real location of iris center) marked manually and the iris center given by the gaze detection method normalized by the distance between the eyes. The GTE_{iris} is given by:



(a) PG: $GTE_{LeftEyelid}$ (b) PG: $GTE_{RightEyelid}$

(c) UImHPG: $GTE_{LeftEyelid}$ (d) UImHPG: $GTE_{RightEyelid}$

Figure 3.24: Comparison between with and without hole eye models of the left (left figures) and right (right figures) eyes on the PG ((a) and (b)) and the UImHPG ((c) and (d)) databases.

$$GTE_{iris} = \begin{cases} \frac{mean(d_{left}, d_{right})}{d_{eyes}} & \text{if } -d \leq R_{yaw} \leq d \\ \frac{d_{right}}{d_{eyes}} & \text{if } -90 < R_{yaw} \leq -22 \\ \frac{d_{left}}{d_{eyes}} & \text{if } 22 \leq R_{yaw} < 90 \\ \frac{\alpha d_{left} + \beta d_{right}}{d_{eyes}} & \text{else} \end{cases} \tag{3.13}$$

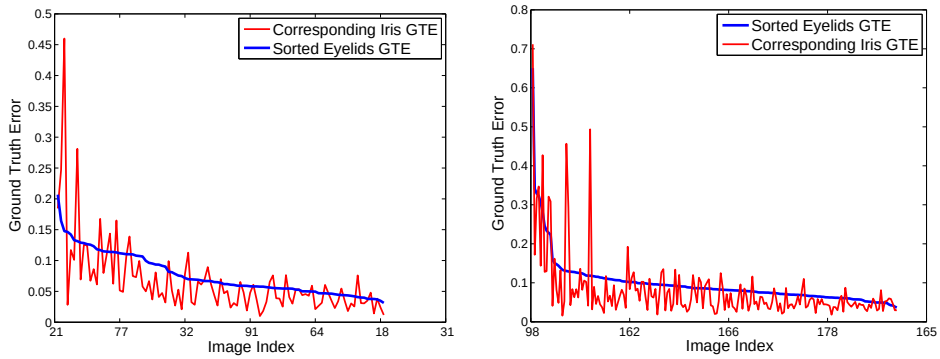
where d_{left} and d_{right} are the euclidean distances between the located eyes and the ground truth, d_{eyes} is the distance between the two eyes from a frontal face.

To study the dependency of the proposed Multi-Texture AAM on the eyelids localization, we plot the $GTE_{2eyelids}$ for each image sorted in decreasing order. We then sort the GTE_{iris} according to the indices of the sorted images of the $GTE_{2eyelids}$. The idea is to see how the GTE_{iris} acts with the decrease of the $GTE_{2eyelids}$. In other words, we can

(a) Comparison of results for the PG database: model without hole (left image) and that of the eye model with hole (right image)

(b) Comparison of results for the UImHPG database: model without hole (left image) and that of the eye model with hole (right image)

Figure 3.25: Qualitative comparison of eyelids model with and without hole



(a) PG database (b) UImHPG database

Figure 3.26: GTE_{eyelid} vs. GTE_{iris} sorted in descending order

assume that if both errors decrease together then they are dependent from each other. Thus a bad eyelid localization would result in a bad gaze detection.

In figure 3.26, we plot the $GTE_{2eyelids}$ of the PG (figure 3.26a) and UImHPG (figure 3.26b) databases sorted in descending order vs. the GTE_{iris}. From this plot, we can see how the GTE of the iris decreases as the $GTE_{2eyelids}$ does. This confirms that if the

localization of the eyelid was precise enough, the MT-AAM will be precise. As a conclusion, we can state that one of the drawbacks of our proposed method is its dependency on the eye localization method. And so, we choose the eye skin model with a hole to locate the eyelids.

3.2.2.2 Comparison between different optimizations

In this section, we compare several optimizations of the iris parameter. This comparison led us to choose the best suited optimization for this parameter. First, different options of the GA are compared. The objective of this comparison is to find the set of options of GA that best suites the parameters in question. The different options that were tried concern the initial chromosomes number, the selection method and the number of chromosomes that will be passed to the next iteration. Next, the best optimization with the best options for GA are compared with Simplex and Gradient Descent (GD).

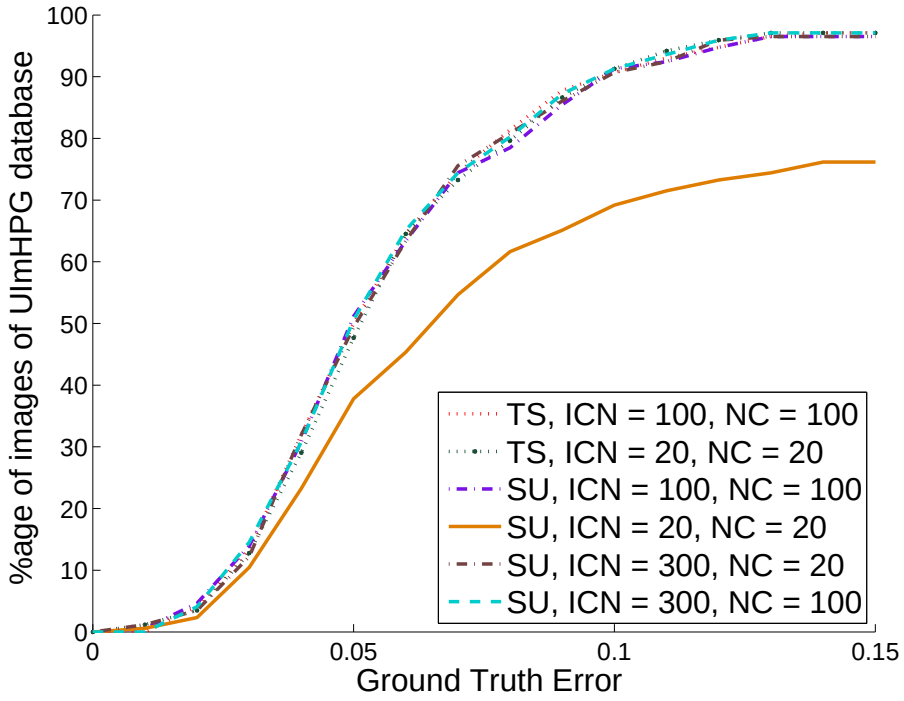


Figure 3.27: Comparison between different options of GA

To do this comparison, we plot the GTE_{iris} in figure 3.27. Table 3.3 presents these different options, together with the computation time corresponding to each of them and the percentage of good detection at 10% of interocular distance. Comparing these values, we find that using uniform selection option with 20 initial chromosome number and 20 chromosomes has the worst performance with a good detection for 69.19% of the test database at 10% of the interocular distance. The tournament selection option, with the same number of chromosomes, performs better with a good detection of 90.28% for the same error. Having the same number of chromosomes, the computation time of the tournament selection is the same as stochastic uniform selection. Keeping the uniform selection and increasing the number of chromosomes to 100 instead of 20 increases the performance

Table 3.3: Comparison of the computation time for the different options of GA. TS = Tournament Selection, SU = stochastic uniform ,ICN = Initial Chromosomes Number, SM = Selection method, NC = Number of Chromosomes, Number of iterations for all is 20 iterations. The line in red is the retained solution.

GA	Options			Computation Time 2 eyes	GTE at 10% of inter-ocular distance
	SM	ICN	NC		
	TS	20	20	$\approx$ **35**	90.7%
	SU	20	20	$\approx$ 40	69.19%
	SU	100	100	$\approx$ 159	**91.28%**
	SU	300	20	$\approx$ 54	90.7%
	TS	100	100	$\approx$ 159	90.7%
	TS	300	100	$\approx$ 187	**91.28%**
	TS	300	20	$\approx$ 54	90.7%

to 91.28% but increases the computation time as well. Increasing the number of initial chromosomes to 300 also increases the performance to 90.7 and the computation time. Increasing the number of chromosomes or the number of initial chromosomes or both for the tournament selection option does not increase the results which means that with this option we arrive at the optimal solution at early stages.

Among all of these options, tournament selection with a number of chromosomes equal to 20 seems to perform the best taking into account the computation time.

Figure 3.28 presents the comparison of the GTE of the best options of GA with Simplex and Gradient Descent (GD) optimizations. As the figure indicates, GA gives the highest GTE curves among these three different optimizations. As a matter of fact, for an error less than or equal to 10% of the inter-ocular distance, we have a good detection of the iris center of 91.28% of the total number of test images in the case of GA versus 75.58% using the GD and 86.63% using the simplex algorithm.

Concerning the computation time of each, table 3.4 compares this for these optimizations. As we see from the table, the Simplex algorithm has the least computation time ($\approx 10sec$ per image) among the others, GD ($\approx 20sec$) takes more time than Simplex but less than GA ($\approx 35sec$). Having the computation time of simplex, the least among all and the % of good detection the second best, Simplex can be a compromise between computation time and accuracy. However, we choose to use the GA because our goal is to acheive the most accurate results.

3.2.2.3 Multi-Objective AAM vs. Single-Objective AAM

In order to test the power of the Multi-Objective AAM (MOAAM), we compare it to a Single-Objective AAM (SOAAM). For this experiment we choose to test on the PG database and not on the UImHPG. Actually, the UImHPG database does not contain con-

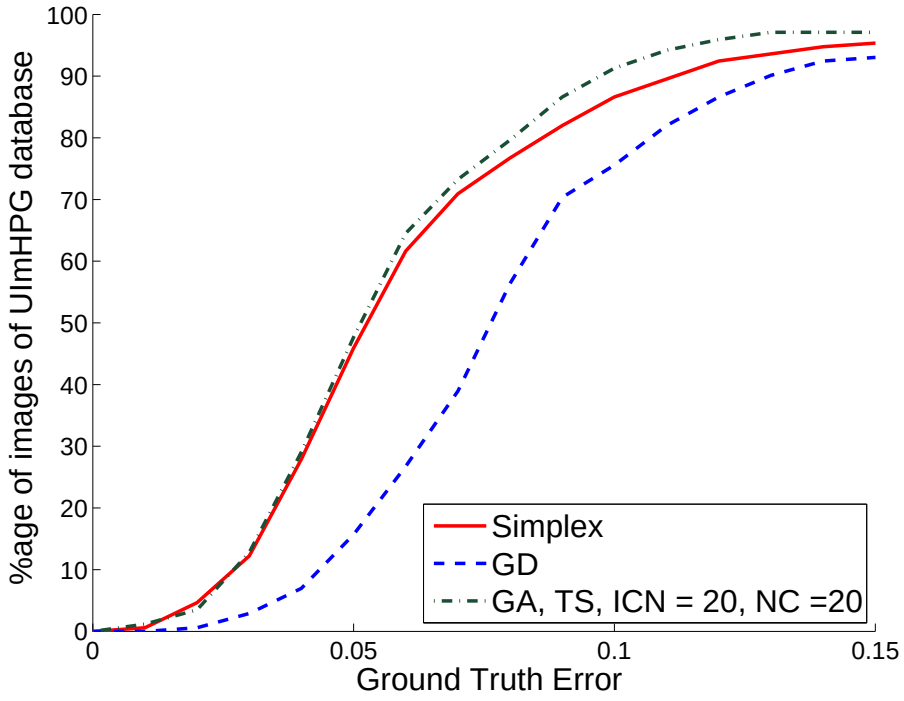


Figure 3.28: Comparison between different optimizations

tinuous head poses (the head poses are increments of 10) in contrary to the PG database. Consequently, we choose the latter since the Multi-Objective method weights the errors according to the head pose. On the other hand, for this experiment, we use the Bosphorous database as a learning base for the eye skin model. In SOAAM, both left and right eyes are given the same weight of 0.5 in the error calculation. In the MOAAM, the proposed double logistic function (cf. section 3.1.3.2) is used to evaluate the weights corresponding to the errors of each of the eyes.

To eliminate the noise of pose detection and to have a fair comparison showing the strength of integrating the head pose in the calculation of the gaze, we use the ground truth of the pose instead of the results given by the global AAM. Tests were done on

Table 3.4: Comparison of the computation time of the different optimizations with the best GA options

Optimization	Computation Time
GA	≈ 35
Simplex	≈ 10
Gradient Descent	≈ 20

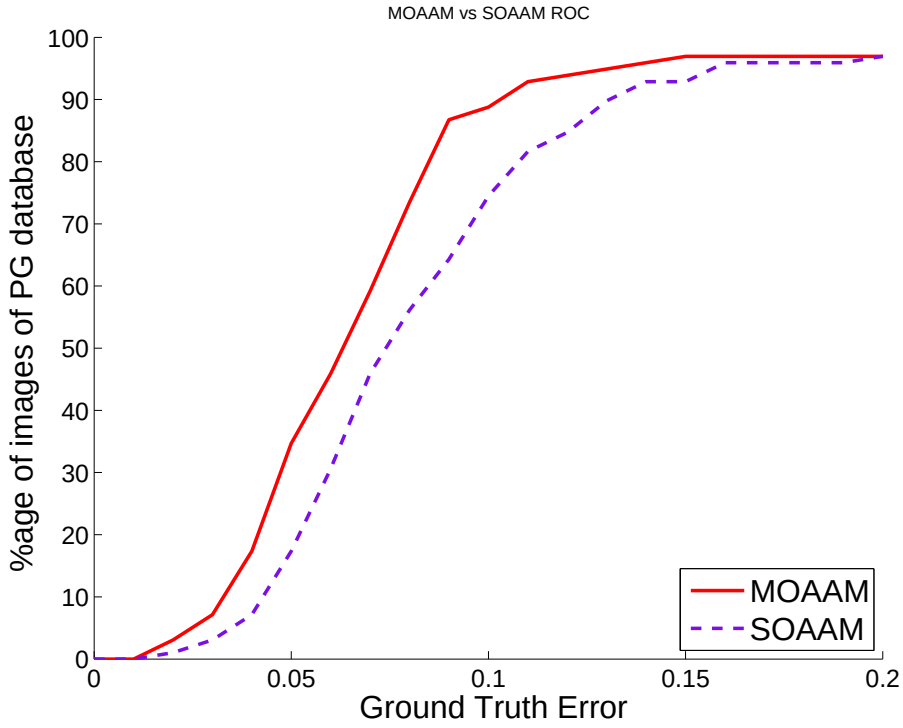


Figure 3.29: MOAAM vs. SOAAM

subjects with a pose in the range of $[-21°, -8°]$ and $[8°, 21°]$ where the information of both eyes is taken into account by the double logistic function. Other poses where not taken into consideration for this experiment since it will not be fair for the SOAAM (for poses $\geq 22°$ one eye is taken into account for MOAAM).

Figure 3.29 shows comparison between the GTE curves of the MOAAM and the SOAAM for the PG database. We can see that MOAAM improves iris localization with respect to SOAAM by 14.29% (iris is well detected for 88.78% of the images with an error level of 10% for MOAAM vs. 74.49% for SOAAM with the same error). As we see, integrating the head pose in the calculation of the final error of the MT-AAM improves iris localization.

3.2.2.4 3D MT-AAM vs. 2D MT-AAM vs. classical AAM

In this section, we compare the $3D$ MT-AAM to a $2D$ MT-AAM and a classical AAM, the $2.5D$ Double Eyes AAM ($2.5D$ DE-AAM). In the following, we describe how the three methods were trained, the optimization methods that were used for each one of them, the constraints applied during search and the results of testing on the PG and UImHPEG databases.

Models creation –

The $2D$ MT-AAM is similar to the $3D$ AAM except that no projection on a sphere takes place where we model the interior of the eye as a simple 2D texture instead of a $3D$ structure. The interest of this approach with respect to the $3D$ one is that it is more efficient for real time applications since it requires less computation time.

Thus, the movements of the iris are modeled using translational parameters instead of rotational ones. The iris pose parameters would then be:

$$T^{iris} = \left[S^{iris} t_x^{iris} t_y^{iris}\right] \tag{3.14}$$

where S^{iris} is the scale of the iris, t_x^{iris} and t_y^{iris} are the horizontal and vertical translation parameters describing the iris points position from the mid-point of the eye.

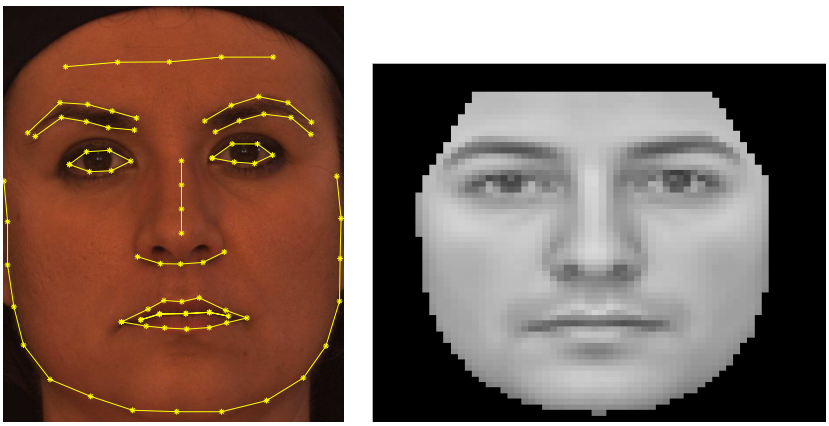
Concerning the fitting using $2D$ representation, it is the same as that of the $3D$ representation except that steps (a)i to (a)iv of the algorithm presented in section 3.1.2.2 are deleted.

The $2.5D$ DE-AAM is built using a total of 28 landmarks. Each eye is annotated by 7 landmarks of which 1 landmark is for its center. To take into consideration the texture surrounding the eyes, landmarks at the bottom of each eyebrow are placed. To train the model, we use 50 images of the PG database as a learning database. It consists of 10 persons with frontal head pose, each committing 5 gaze directions (1 to the extreme left, 1 to the extreme right, 1 to the front and 2 intermediate gaze directions).

For the $2D$ MT-AAM and the $3D$ MT-AAM, we use 10 persons of the PG database looking in front of them to train the eye skin model. The images are the same as those used for the training of the $2.5D$ DE-AAM but without the different gaze directions. The iris AAM is trained using a group of 23 iris textures starting from the images of [DMTP04] (see section 3.1.2.1). Figure 3.30 shows the set of iris images used to train the iris AAM. Concerning the two MT-AAMs, to find the head orientation, a $2.5D$ global active ap-

Figure 3.30: Set of iris textures used to train the iris model

pearance model is used. The model is trained on 104 neutral face images of the $3D$ Bosphorous database of [SAD+08b]. A total of 83 landmarks are used, of which 78 are marked manually on the face and 5 landmarks on the forehead estimated automatically from the landmarks of the eyes. We show the annotations of one image of the learning database and the mean texture of the model in figure 3.31.



(a) Annotations (b) Mean texture

Figure 3.31: On the left, the annotations to obtain the head pose model. On the right, the corresponding mean texture

Optimization – Concerning the $2D$ MT-AAM, a Genetic Algorithm (GA) followed by a Gradient Descent (GD) is used to optimize the iris appearance and pose parameters. Concerning the $3D$ MT-AAM, only a GA is used for optimization. Concerning the DE-AAM two consecutive Newton gradient descent algorithms are used. In the first one the learning of the relationship between the error and the displacement of the parameters is done offline during the training phase as proposed by Cootes [CET98c]. In the second, this relationship is learned online. The parameters from the first optimization scheme are entered into the second in order to refine the results.

Constraints – Since the iris location and scale are constrained by the size of the eye, constraints are added in order to tighten up the search space of the iris pose vector for the 2 versions of MT-AAM.

2D MT-AAM – Constraints are based on anthropometric averages which give a very tight bound on the size of the iris relative to the size of the eye and on the fact that iris movements have limits imposed on them by the width of the eye. Iris average width is approximated at 1/1.6 the width of the eye.

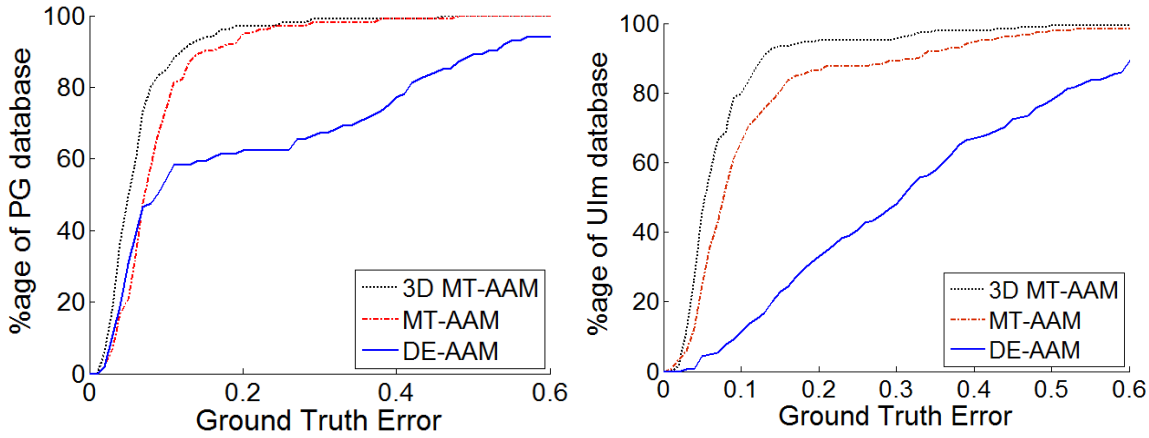
3D MT-AAM – The horizontal rotation of the sphere is limited to $+40°$ and $-40°$ and the vertical rotation is limited to $+10°$ and $-10°$ which is found to give plausible projections of the $3D$ iris shape.

For both $2D$ and $3D$ MT-AAM, the scale is varied around an initial scale calculated using the width of the iris and that of the mean iris. The horizontal and vertical translation parameters cannot exceed half of the eye width and height respectively, taking the midpoint of the distance between the eye corners as the origin point.

Testing – We conduct two types of tests. The first test was made on a sample of images from the PG database. This means that we test on images coming from the database that was used for training the models. Of course testing was done on another set of images

than the learning one. The second test was made on the UImHPG database. This test suggests generalization.

The UIm testing database contains 185 images chosen randomly from the initial database. The PG testing database contains the same persons of the training database (mentioned in the paragraph concerning training) but with varying head poses and gaze directions. The number of images in it is 100 of which are chosen randomly from the initial database.



(a) GTE of PG database (b) GTE of UImHPG database

Figure 3.32: 3D MT-AAM vs. 2D MT-AAM vs. Double Eyes AAM

In figure 3.32, we compare the Ground Truth Error of the iris (GTE_{iris} presented in section 3.2.2.1.2) versus the percentage of aligned images by testing on the Pose Gaze and the UImHPG databases. Both figures contain 3 curves (GTE_{iris} for the 3 AAM versions).

2D and 3D MT-AAMs vs. DE-AAM – As we see from figure 3.32a, both the 2D and 3D MT-AAM outperform the 2.5D DE-AAM. For instance, for an error less than or equal to 10% of the inter-eyes distance, the 3D MT-AAM has detected the correct position of the iris on 85.15% of the images, the 2D MT-AAM has detected 74.26% whereas the 2.5D DE-AAM shows only 54.46% at the same error level.

Concerning figure 3.32b (generalization test), it shows that the MT-AAM model outperforms the classical one in generalization. Actually, as we separate the interior of the eye from the eye skin, we are able to parametrize the iris location through the iris pose vector (cf. section 3.1.2). Thus, it does not enter in the AAM appearance parameters anymore. Consequently, we are able to generalize to new different people while being less dependent from the learning base. This proves that even when the testing database contains the same persons as the learning database for the DE-AAM, the multi-texture AAM overcomes the latter.

For the DE-AAM, when the person is in frontal view, the model succeeds to localize the iris which is normal because the learning was done on such kind of images. However,

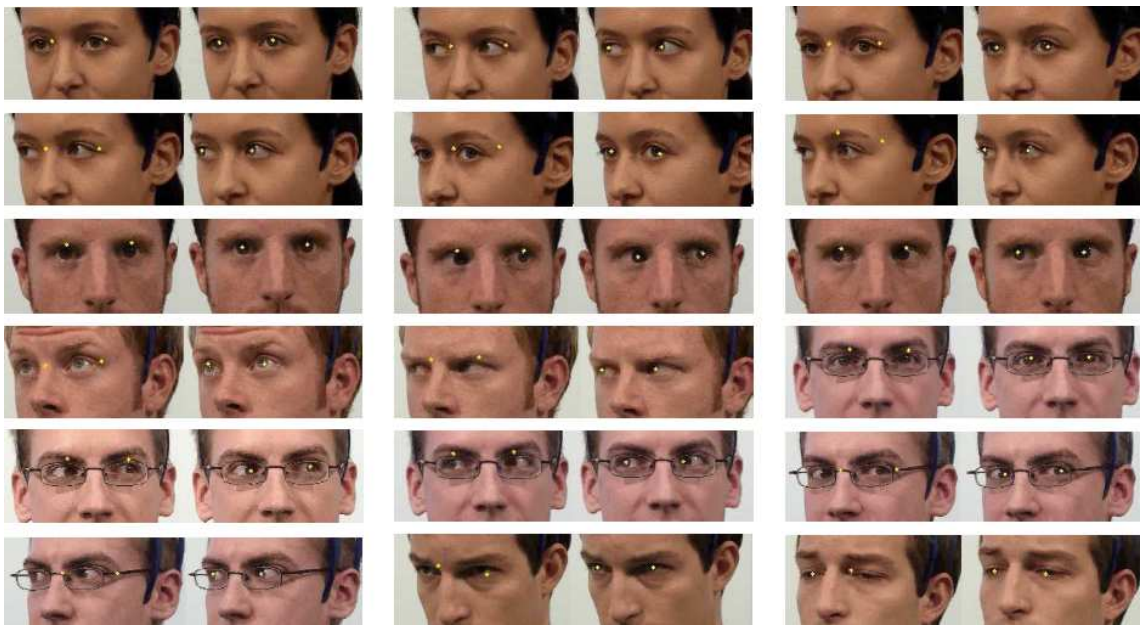


Figure 3.33: Qualitative comparison between the $2D$ multi-texture approach (right columns) and the DE-AAM approach (left columns).

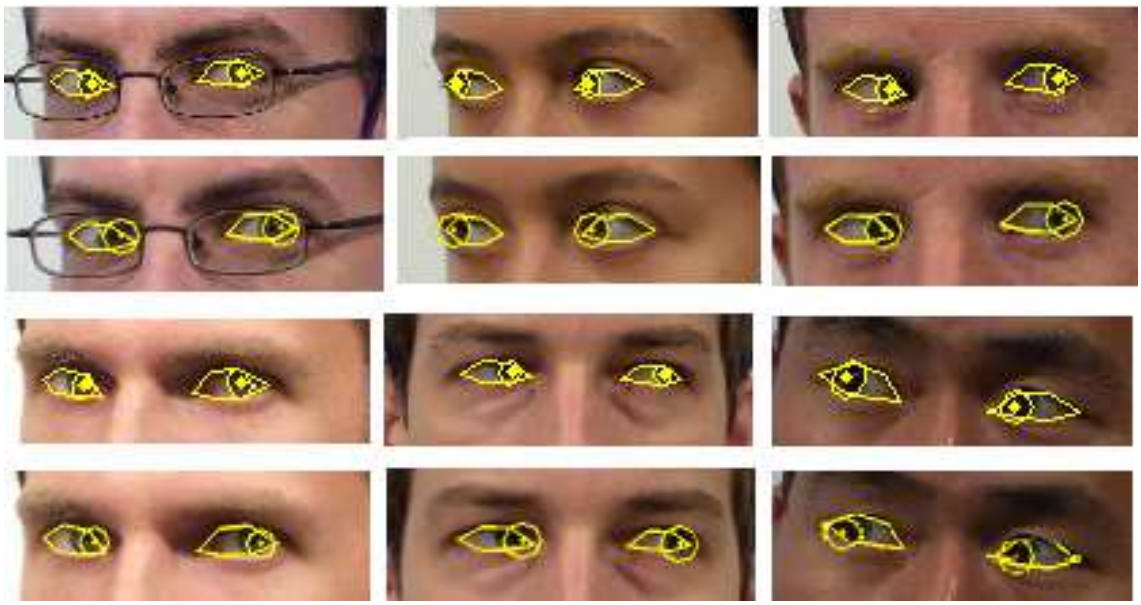


Figure 3.34: Qualitative comparison between the $3D$ MT-AAM (upper row of the same person) and the $2D$ MT-AAM (lower row of the same person).

when it comes to the different head poses present in the testing database, the DE-AAM will fail while the MT-AAM will not. This is due to the fact that in the DE-AAM it is necessary to increase the number of images in the learning database to increase the accuracy and to include variation in pose. Whereas, with the MT-AAM, the number of images in the learning database is sufficient to localize accurately the eyelids since the interior is removed and we have the same persons; then for the iris location, the general iris texture that is slided under the skin is able to localize it.

2D vs. 3D MT-AAM – As we see from figure 3.32a, the 3D MT-AAM outperforms the 2D MT-AAM which confirms the success of modeling the iris as a part of a sphere which is normal because it is more realistic. Concerning figure 3.32b (generalization test), we have a good detection of 80% for the 3D MT-AAM versus 65.95 for the 2D MT-AAM and 11.35% for the 2.5D DE-AAM method for the same error level of 10%. This confirms first that the 3D MT-AAM is also better than the 2D MT-AAM in generalization.

We would like to point out that modeling the iris as a part of a sphere instead of a plane permits to give the gaze angle directly. This shows another advantage of a 3D representation over the 2D one. In the latter, the pose of the iris is in terms of vertical and horizontal translations. Thus, to compute the gaze angle, further calculations should be done. We compare these two representations in the results section of this chapter and show the superiority of the 3D representation over the other.

In addition, figure 3.33 shows qualitative results on the UImHPG database to compare the MT-AAM and the DE-AAM. As the figure shows, MT-AAM succeeds to follow the gaze of persons with eye glasses and with different head poses, whereas the DE-AAM method does not. This assures the fact that with our method we are able to restrict the training base where there's no need to include people wearing eyeglasses in order to get reliable results on such subjects. The reason behind this is that as we divide the eye model into two models (eyes skin and iris), the iris model is not disturbed by the glares of the glasses.

On the other hand, figure 3.34 shows some qualitative results comparing the two models: the 3D MT-AAM and the 2D MT-AAM. In addition to the iris center, we show the iris shape on these images to show the difference between these two models in this aspect. It is obvious from the figure the superiority of the 3D AAM on the 2D AAM. As we see for extreme gaze directions the iris in the 3D MT-AAM will take the shape of an ellipse representing realistically its appearance, however for the 2D MT-AAM, the iris shape is a circle that comes out of the eye in most of the times because it can not take the real shape of the iris. As a conclusion, we can state that the 3D representation of the iris is more realistic than a 2D one and thus gives better results.

3.2.2.5 Comparison with a state-of-the-art method

This part compares the 3D MT-AAM method to the state-of-the-art method of [HSL11]. The comparison is conducted on 3 image sequences of the UImHPG database (heads 3, 12

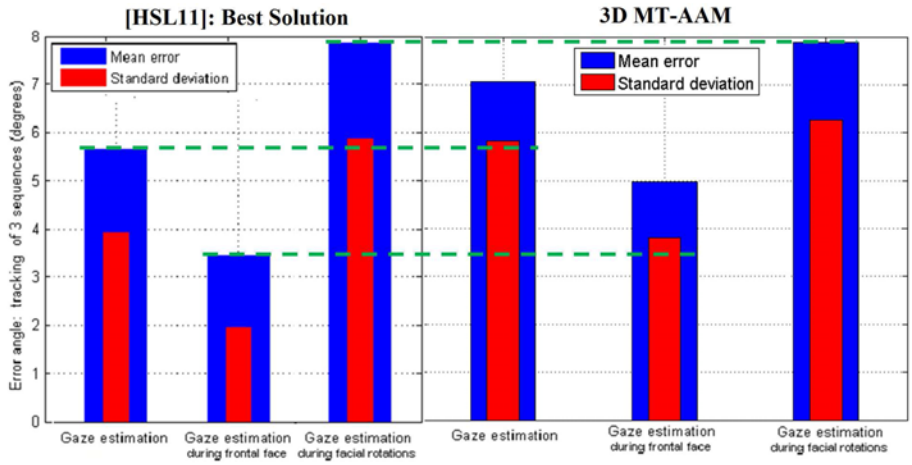


Figure 3.35: Comparison of the 3D MT-AAM (right) method to that of [HSL11] (left)

and 16). The authors perform 2 trackings of the gaze of each of these sequences using 2 slightly different manual initializations of their head model. Head rotations were restricted to $[-30°, 30°]$ and gaze rotations to $[-40°, 40°]$. On the other hand, we only conduct one gaze detection experiment per sequence since our method is fully automatic.

Figure 3.35 shows the average error angle of our method compared to that of [HSL11]. The red color corresponds to the standard deviation and the blue color corresponds to the mean of the gaze angle. We achieve a 7.07° gaze angle error with our method compared to a 5.64° with their method. We achieve their accuracy in the case of facial rotations. In the case of frontal face, their method is better than ours by about 1.5°.

The authors have more accurate results because they manually initialize their first frame for tracking. Thus, the authors guarantee that they will not have high $GTE_{2eyelids}$ as we do (cf. figure 3.26; when we have high $GTE_{2eyelids}$, we have high GTE_{iris}, i.e. a bad gaze detection). As a conclusion, since our method is fully automatic and we acheive similar results in the case of facial rotations, we can say that our method is more robust and more appropriate for real time applications.

3.3 Conclusion

In this chapter, we have presented the Multi-Object Facial Actions Active Appearance Model. The model subtracts eye motions from the appearance parameters of AAM and puts them in an independent vector of parameters. This has the advantage of restricting the database of AAM into neutral-faces subjects. The multi-object representation suggests dealing with the face as a combination of several objects. Objects to be included in the system are specified according to the case in question. This has been integrated in the context of a gaze detection application. The different objects are the two eyes and the

head pose plays the role of the criterion that specifies which object contributes the most in the gaze detection system.

Chapter 4

Face modeling for emotion recognition

Sommaire

The face is the key to understanding emotion, and emotion is the key to understanding the face.

– J. A. Russel, and J.M. Fernandez-Dols *The psychology of facial expression*

The nature of the Human-Human interaction is multi-modal. When two people interact with each other, they interact through their faces (in the form of facial expressions),

through their voices (tonality), through their conversations (spoken words), body movements and posture, hand gesture, head pose, and their eyes (gaze). Through this interaction, they communicate their internal emotions. This makes evidence that the nature of emotion perception is multi-modal.

Perhaps facial expression is the cue that reveals emotion the most because a person expression is easier to interpret than the other cues and because expressions are closely tied to emotion. For example, if a person is happy, he will tend to smile without a doubt. This is why automatic recognition of human emotion has focused on facial and vocal emotion interpretation in terms of six basic emotions [E+93] which are tied to particular facial expressions: neutral, sadness, happiness, fear, anger, surprise and disgust. Facial AUs [EF77] detection was also abundantly used in a categorical approach of emotion interpretation. However, researchers have argued that the interpretation of emotion with respect to six limited independent emotions is not enough to describe the complex and subtle nature of emotion.

This gave birth to other axes of emotion interpretation: the dimensional approach and the appraisal-based approach. The dimensional approach interprets emotions in terms of some continuous dimensions in an affect space. In this approach emotions are not independent from each other and relate to one another by these dimensions.

The Facial Expression Recognition and Analysis Challenge (FERA 2011) and Audio/Visual Emotion Challenge (AVEC 2012) are two challenges that have one of their objectives to provide a common benchmark testing set for emotion detection. FERA 2011 considers the categorical approach of emotion detection where two sub-challenges where organized concerning discrete emotions and AUs detection. Only video data where considered, thus the challenge is uni-modal. On the other hand, the AVEC 2012 generalizes to the use of multi-modal cues in the detection of emotion using the dimensional approach.

These challenges were a chance for us to test our feature detection skills using Active Appearance Models on real naturalistic data. They also permit us to employ our feature detection capabilities in automatic emotion detection which is a very active and important subject to Human Computer Interface and human psychology.

In the following sections, we present each of the two challenges and the data that we worked on. We describe the systems that we have employed and how we have adapted our skills in facial features detection using AAMs in these challenges. We specify our contributions at the levels of the systems creation and facial features detection. Section 4.1 is dedicated to the FERA 2011 challenge and section 4.2 describes the FERA 2012.

4.1 The Facial Expression Recognition and Analysis Challenge

The Facial Expression and Analysis challenge (FERA 2011) [VJM+11] is the first challenge in automatic recognition of facial expressions organized in conjunction with the IEEE International conference on Face and Gesture Recognition 2011. The objective of

the challenge is to provide a common data set where participants compete and compare their methods.

The challenge is divided in two sub-challenges that reflect two popular approaches to facial expression recognition: an AU detection sub-challenge and an emotion detection sub-challenge. The AU detection sub-challenge calls for researchers to attain the highest possible F1-measure for 12 frequently occurring AUs (cf. figure 4.1). The emotion detection sub-challenge calls for systems to attain the highest possible classification rate for the detection of five discrete emotions: anger, fear, joy, relief, and sadness.

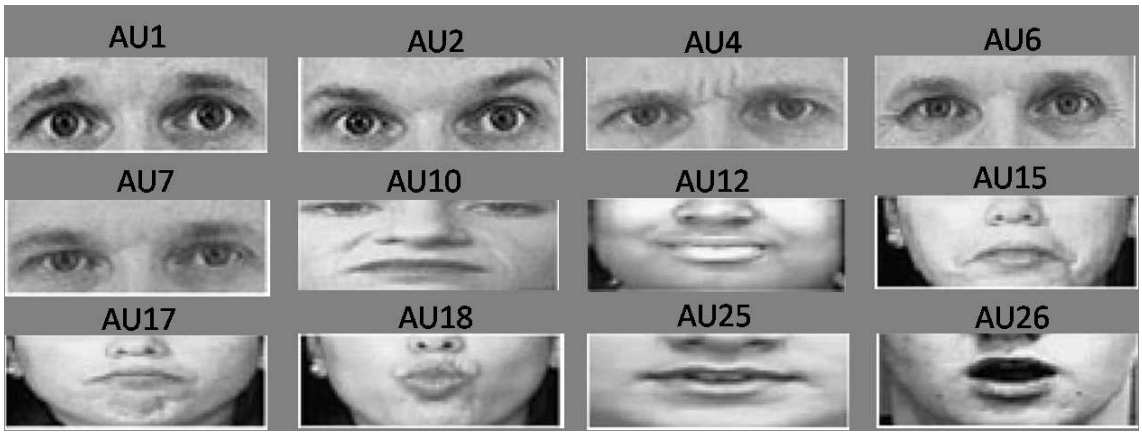


Figure 4.1: The Action Units to be detected for the FERA 2011 challenge

In the framework of **IMMEMO** ANR project (**IMM**ersion 3D basée sur l'interaction **ÉMO** tionnelle) [IMM], we have participated in the AU detection sub-challenge in collaboration with the ISIR, LAMIA and LTCI teams and took the first place [SRS+11]. The proposed method was later tested for emotion detection and gave results that are very close to those who won the second sub-challenge [YB11].

The challenge uses a partition of the GEMEP corpus [BS07] developed by the Geneva Emotion Research Group (GERG) at the University of Geneva led by Klaus Scherer. This database presents a number of professional actors performing 18 emotions. For the AU sub-challenge, the database is split into two partitions: a training partition consisting of 87 videos and a test partition consisting of 71 videos. For the emotion sub-challenge, a total of 289 videos were selected (155 for training and 134 for testing). For both sub-challenges, the training set includes 7 actors and the test set includes 6 actors, half of which were not present in the training set.

The difficulty of this database is that the expressions displayed by the actors are spontaneous and natural. Speech is included which results in more variability in the lower part of the faces in the database (actors pronounce meaningless phrases or the word "aaah"). In addition, actors are not posed, in contrary they act naturally and move their faces freely. Figure 4.2 shows some examples of the images of this database.

Figure 4.2: Examples of some images of the GEMEP-FERA dataset

4.1.1 System overview

AU detection can be classified into three categories: appearance feature-based [BLF+06], geometric feature-based methods [SLS+07, VP06] and hybrid features methods. Appearance feature-based methods extract features that try to represent the facial texture such as wrinkles, bulges and furrows. Geometric-based methods extract the facial shape and the location of facial points. Hybrid methods combine both features [CLL+11a, ZJ05a].

We propose an AU detection system that belongs to the hybrid features family. It combines appearance features in the form of Local Gabor Binary Pattern histograms and geometric features in the form of Active Appearance Models appearance parameters.

In the following, first we briefly describe the overall system of AU detection proposed by our team, along with the different techniques used (section 4.1.1.1). Then we detail our contribution at the level of extraction of AAM coefficients (section 4.1.2). Finally, we present the results of the challenge and we show how the AAM coefficients contribute to increasing the overall results (section 4.1.3).

4.1.1.1 Hybrid-features Action Unit detection system

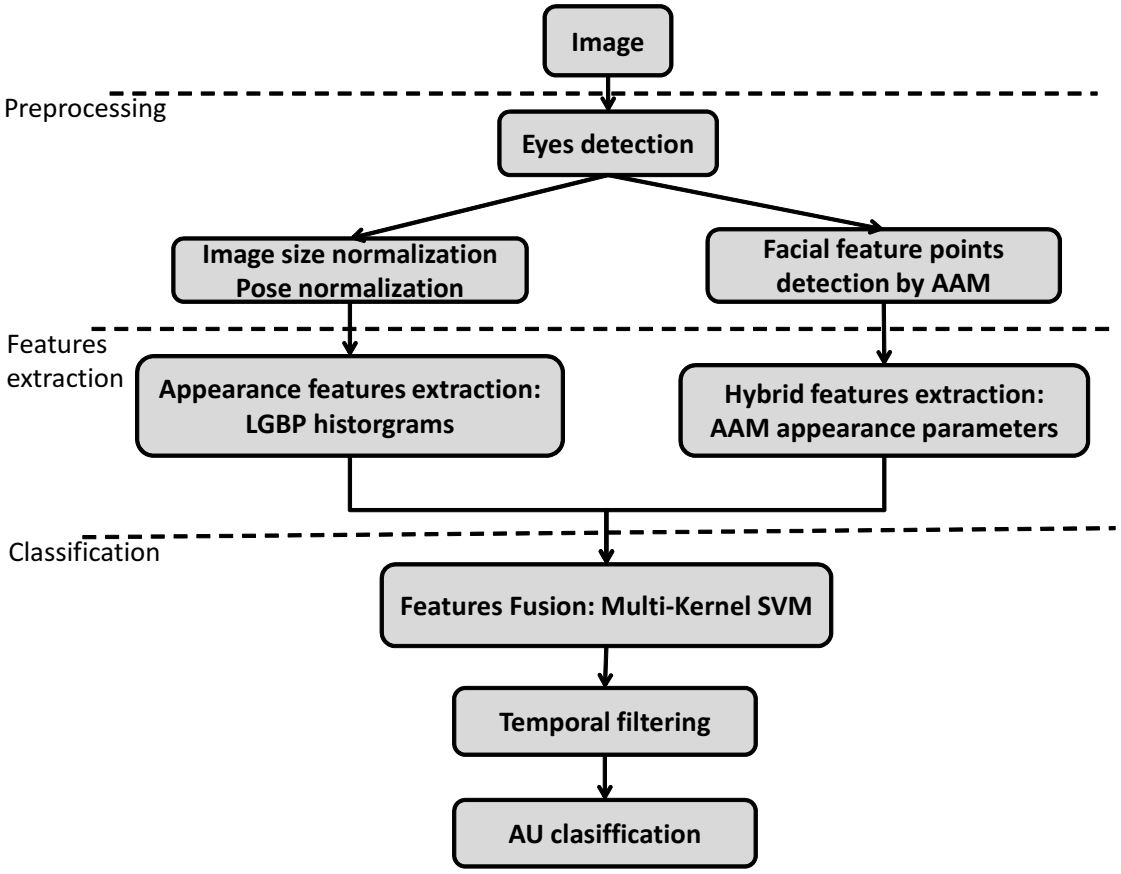


Figure 4.3: Global system of AU detection

Automated facial expression recognition is generally accomplished by four steps:

1. Preprocessing and that includes face detection and landmark localization;
2. Image coding, and that is feature vectors extraction;
3. Classification of the image as a positive sample (an AU, several AUs or an emotion is present in the image) or a negative sample;
4. Temporal analysis;

Figure 4.3 presents the flow chart of our global system for AU detection.

Preprocessing – First eyes are detected using the feature localizer of [RSBP11] based on multiple kernel SVM learning. Eyes centers are then used to: 1) normalize the image with respect to scale, position and in-plane rotations variations and with respect to the image size. This is used for the appearance-based features extraction; 2) Apply Active

Appearance Models to the test images to extract the facial feature points (see the following paragraph).

Features extraction –

Two types of features are extracted in our AU detection system: Appearance features in the form of Local Gabor Binary Pattern (LGBP) histograms introduced by [ZSG+05] and hybrid features in the form of Active Appearance Models (AAM) coefficients (AAMs combine geometric and appearance features).

Appearance features: LGBP histograms – The normalized images are used to calculate LGBP histograms. The interest of such features is that they exploit the multi-resolution and the multi-orientation links between pixels. In addition, they are known to be robust to illumination changes and misalignments. LGBP are computed in the following manner.

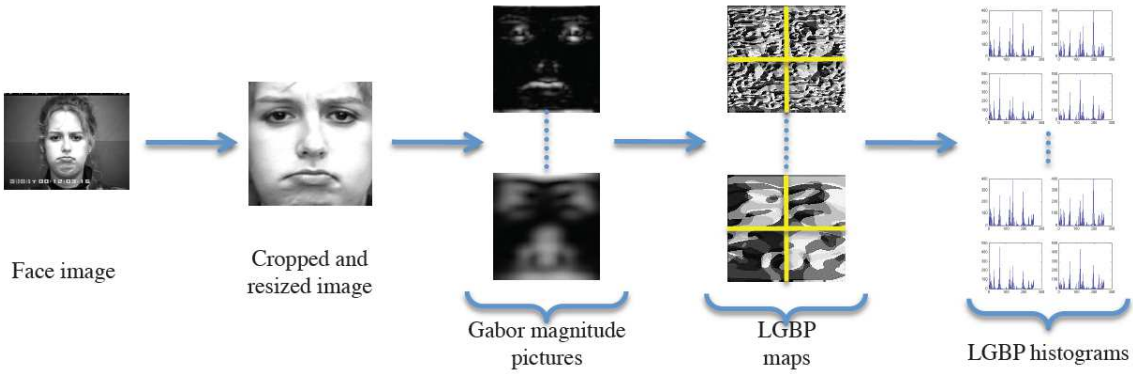


Figure 4.4: Local Gabor Binary Pattern histograms computation

1. Compute the Gabor magnitude pictures by convolving the facial image with Gabor filters: Three spatial frequencies were used and six orientations for a total of 18 Gabor filters. Only the magnitude was kept since the phase is very sensitive. This results in 18 Gabor magnitude pictures.
2. Compute the Local Binary Pattern (LBP) for each of the 18 Gabor magnitude images.
3. Divide the face into $n \times n$ areas and compute a histogram per area. This accounts for the different useful information for the AUs contained in the different facial regions.
4. Concatenate all the histograms in the feature vector H_i after reducing the number of bins in each histogram [SRS+12].

Hybrid features: AAM coefficients – The extraction of AAM coefficients is presented in section 4.1.2. The importance of AAM for this system is that they can provide important spatial information of key facial landmarks. Thus, it compensates the loss of spatial information which really depends on identity when using just LGBP histograms.

Classification – The proposed AU detection system uses Support Vector Machines (SVM) for their ability to find an optimal separating hyper-plane between the positive and negative samples in binary classification problems. As both features (LGBP histograms and AAM coefficients) are very different, they were not concatenated in a single feature vector. Instead of using one single kernel function, two different kernels were used, one adapted to LGBP histograms and the other to AAM coefficients. These kernels were combined in a multi-kernel SVM framework [RBCG08]. In our case, we have one kernel function per type of features.

$$K = \beta_1 K_{LGBP}(H_i, H) + \beta_2 K_{AAM}(C_i, C) \tag{4.1}$$

β_1 represents the weight accorded to the LGBP features and β_2 is the one for the AAM appearance vector. Thus, using a learning database, the system is able to find the best combination of these two types of features that maximizes the margin.

K_{LGBP} and K_{AAM} are the kernel functions for the LGBP histograms and the AAM coefficients respectively where for the LGBP histograms, histogram intersection kernel was used and for the AAM appearance vectors, the Radial Basis Function (RBF) kernel was used:

$$K_{LGBP}(H_i, H_j) = \sum_k min(H_i(k), H_j(k)) K_{AAM}(C_i, C_j) = e^{-\frac{\|s_i - s_j\|_2^2}{2\sigma^2}} \tag{4.2}$$

With σ a hyper-parameter we have to tune on a cross-validation database.

This is a new way of using multi-kernel learning. Instead of combining different kinds of kernel functions (for example Gaussian radial basis functions with polynomial functions), we combine different features.

The AAMs modeling approach takes the localization of the facial feature points into account and leads to a shape-free texture less-dependent to identity. However, one of the severe drawbacks is the need of a good accuracy for the localization of the facial feature points. The GEMEP-FERA database contains large variations of expressions that sometimes lead to inaccurate facial landmarks tracking. In such cases, multi-kernel SVMs will decrease the importance given to AAM coefficients.

4.1.2 Active Appearance Models coefficients

Even though we have worked on the global system with the team, our principal contribution focused on AAM coefficient generation.

4.1.2.0.1 AU detection: $2.5D$ AAM local models

As AUs are elementary muscle movements that affect small different regions in the face, and due to the fact that the AUs to detect are divided into two parts: the upper AUs and the lower ones, we decide to employ two $2.5D$ AAMs [SALGS07b] local models: one for the mouth and one for both eyes.

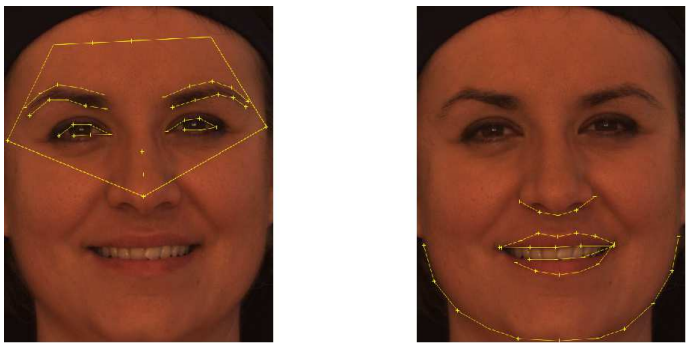


Figure 4.5: Landmarks for the eyes and mouth models

In local models, the shapes and textures of the eyes are not constrained by the correlation with the shape and texture of the mouth, and thus, local AAM's are supposed to give more precise results than a global one for local areas. More precisely, if the testing image does not present the same correlation between the eyes and the mouth as the ones present in the learning base, based on our experiments, a global model will probably fail to converge while the local one will not.

Training – The mouth sub-model is formed of 36 points which contain points from the lower part of the face and from the nose shape. The eyes sub-model contains both eyes and eyebrows with 5 landmarks on the forehead which are estimated automatically from the landmarks of the eyes resulting in 42 landmarks (Fig. 4.5).

Testing – To optimize the appearance parameters of both eyes and mouth AAMs, we employ two consecutive Newton gradient descent algorithms. The difference between the two is that in the first one the learning of the relationship between the error and the displacement of the parameters is done off-line during the training phase as proposed by Cootes [CET98b], while in the second one we evaluate this relationship on line. The obtained parameters from the first optimization scheme are entered into the second in order to refine the results.

4.1.2.0.2 Emotion detection: Global skin model

For emotion recognition we use a different model to obtain the AAM coefficients. We train a global model and not two local models as it is the case for the AU detection part. After trying the local models and a global model for the AU recognition (a global model was applied for the original results of the FERA challenge [SRS+11]), we observed that the interior of the eyes and that of the mouth add misleading variations to the AAM. Actually, the appearance of the eye undergoes many variations due to the scale, color, position of the iris and that of the mouth changes where the teeth and tongue may appear or disappear as the person speaks. This can be confirmed by the tests that compare

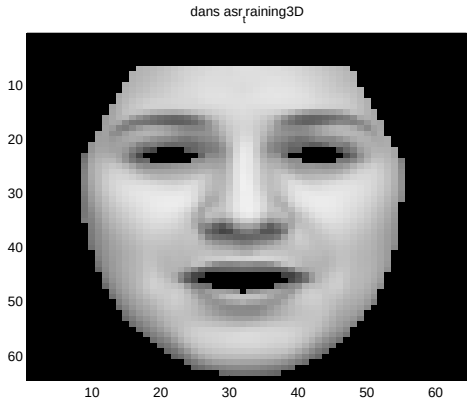


Figure 4.6: Mean texture of the global skin model: holes are put in place of the interior of the eyes and the mouth.

the eyes models with and without holes in the previous chapter (section 3.2.2.1.1, figure 3.24). Consequently, when the interior of the eyes and the mouth are learned with the model, this increases the number of appearance parameters and introduces unnecessary variations. So, we decide to remove their perturbation to the appearance of the face by putting holes in their place while training the model. We call this model the global skin AAM, since only facial skin information was used.

Training – We use a total of 83 landmarks to train this AAM. Figure 4.6 is an illustration of the mean texture of this model.

Testing – To obtain the test appearance parameters, for every sequence in the FERA emotion database, we launch the tests in tracking mode. For every first image of every sequence, force is put to have a good fitting result and then for the rest of the images in the sequence, the model uses the appearance parameters of the preceding image and adapts itself to the current image with less number of iterations. As for the optimization, the same scheme as that with the AU recognition model is used.

4.1.3 Results

4.1.3.1 AAM results

AU detection – AAM local models were trained on a total of 466 expression and neutral images from the Bosphorous $3D$ face database [SAD+08a]. This suggests a pure AAM generalization. Figure 4.7 shows some results of the mouth and local eyes AAM fitting. As we see, for all the images, the localization of the mouth is successful which confirms the efficiency of using a local model for the mouth. Concerning the eyes localization by the eyes local AAM, the figure shows that all the eyes of the images are precisely localized except for the last image. The reason is that for this image the eyes sub-model takes into

		LBP (baseline)	LGBP	Eyes AAM	Mouth AAM	Both AAMs	LGBP +Eyes AAM	LGBP +Mouth AAM	LGBP +Both AAMs
upper AU	AU1	79.0	78.8	60.5	54.4	62.3	77.6	81.8	80.3
	AU2	76.7	77.1	57.4	51.3	57.4	57.0	83.4	82.7
	AU4	52.6	62.9	62.0	56.4	59.3	62.4	61.3	58.5
	AU6	65.7	77.0	56.4	75.8	80.9	79.3	80.9	81.0
	AU7	55.6	68.5	67.8	54.3	54.4	72.5	71.0	71.2
lower AU	AU10	59.7	51.9	43.2	56.3	51.9	49.9	52.7	52.1
	AU12	72.4	79.9	63.9	69.9	79.5	81.7	82.3	82.2
	AU15	56.3	63.0	58.3	68.7	71.2	67.1	59.6	61.4
	AU17	64.6	65.8	51.9	67.1	66.3	67.2	72.5	70.7
	AU18	61.0	70.4	51.9	75.3	75.8	54.8	79.0	78.5
	AU25	59.3	59.8	55.6	69.6	63.6	56.0	63.5	65.6
	AU26	50.0	64.3	57.0	67.5	58.6	58.7	64.8	62.9
Avg person-specific		63.1	67.8	57.4	63.6	65.5	65.5	71.6	71.5
Avg person-indep.		61.1	68.0	57.4	65.3	66.1	65.9	69.5	69.0
Avg overall		62.8	68.3	56.8	63.9	65.1	65.3	71.1	70.6

Table 4.1: 2AFC scores on the GEMEP-FERA test dataset using LGBP, eyes AAM coefficients, mouth AAM coefficients, concatenation of the eyes AAM coefficients and the mouth AAM coefficients and the fusion of LGBP with AAM coefficients.

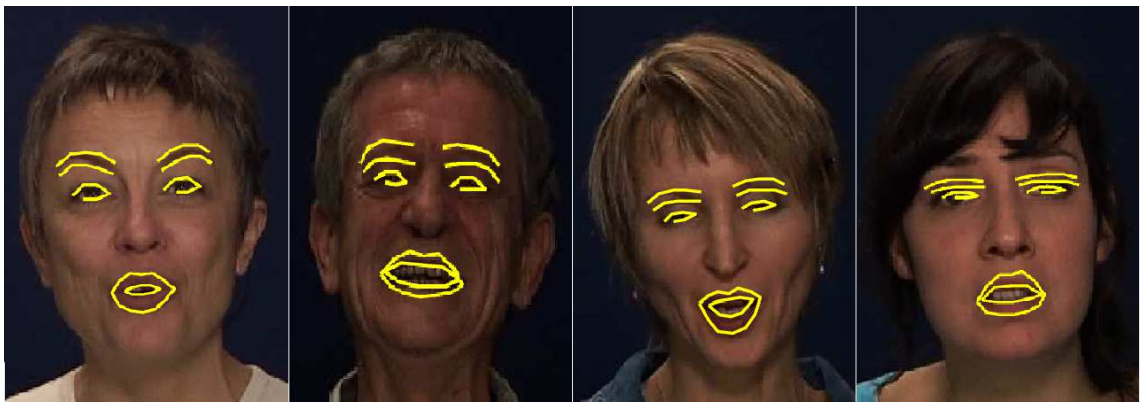


Figure 4.7: AAM local models results on some test images showing successful eyes and mouth segmentation except for the last image where hair perturbation mislead the eyes model.

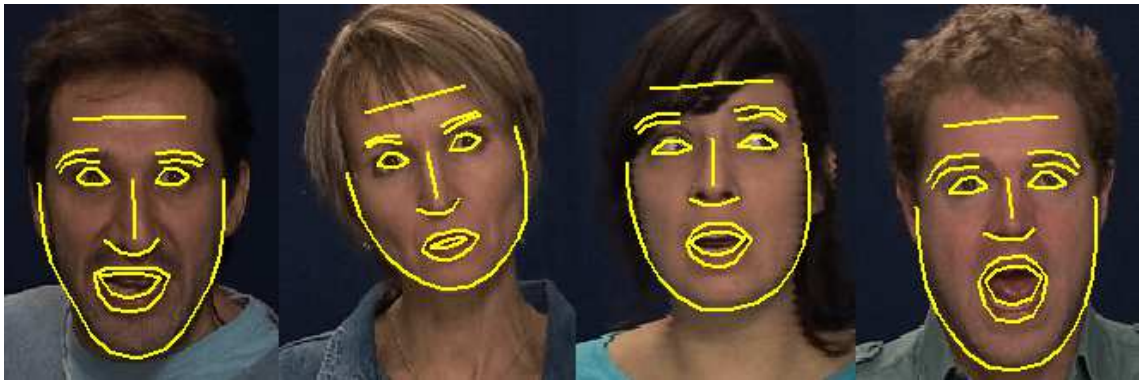


Figure 4.8: AAM global skin model results on some test images showing successful eyes and mouth segmentation. See how for the third image, the hair perturbation does not affect the eyebrows localization taking advantage of a global model and a holes approach

consideration the forehead texture, cf. figure 4.5, which is covered with hair (perturbation) in this testing image. A local model would work in such a case only if the training database contained subjects having same kind of hair on their foreheads. However, this is not the case here. This shows one disadvantage of local models with respect to global ones. As a matter of fact, in a global model the amount of error becomes relatively smaller in local areas having perturbations. In addition, a global model will make use of correlations between the different parts of the face and thus it will have more tendency than a local model to converge in such cases. This was shown in the context of AVEC 2012 challenge that we later participated in.

Emotion detection – The global skin AAM was trained on 598 images from the Cohn Kanade and the Fera databases (258 images). Although it is not our objective to compare the behavior of the global skin AAM and that of local models, it seems interesting to show the result of the global skin AAM on the same subject where the local eyes AAM did not work. Figure 4.8 presents this result together with some results on other images.

4.1.3.2 IMMEMO team challenge results

For the experiments, the following databases are used as training databases:

- The Cohn-Kanade database [KCT00]: the last image (expression apex) of all the 486 sequences. It contains images sequences of 97 university students ranging from ages of 18-30. The lighting conditions and context are relatively uniform and the images include small in-plane and out-plane head motion.
- The Bosphorus database [SAD+08a]: around 500 images chosen because they exhibit a combination of two AUs. The lighting conditions and context are relatively uniform and the images include small in-plane and out-plane head motion.
- The GEMEP-FERA training dataset [BS07]: one frame from every sequence for every AU combination present resulting in 600 images.

The SVM slack-variable and the RBF kernel parameter were optimized using a 7-fold subject independent cross-validation on all the GEMEP-FERA dataset. All the images of one subject from the GEMEP-FERA training dataset are used as a test set (around 800 images). Images of the other subjects within the 600 selected from the GEMEP-FERA training dataset, and eventually from other databases, are used as a training dataset.

4.1.3.2.1 Effect of combining features – In this section, we prove that fusing LGBP histograms with AAM coefficients gives the best results for the AU detection. To do this, we report the area under the ROC curve for the different features and their combinations: by using the signed distance of each sample to the SVM hyper-plan and varying a decision threshold, we plot the hit rate (true positives) against the false alarm rate (false positives). The area under this curve is equivalent to the percentage of correct decisions in a 2-Alternative Forced Choice task (2AFC) by which the system must choose which of the two images contains the target.

The system is trained on the GEMEP-FERA training dataset. We compare the 2AFC of our system to that of the baseline for: person independent (test subjects are not in the training database), person specific (test subjects are in the training database) and overall. We report in Tab. 4.1 the overall results for each AU and the average for the person specific, person independent and overall case.

Using only LGBP, we notice that we already have better results than the baseline proposed by the organizers (68.3% against 62.8% overall). The two methods are really similar: equivalent setup, same training database, same classifier, only the features and the kernel function of the SVM are different.

Using only the AAM appearance vector, we notice that we have good results using the mouth AAM to detect the AUs localized in the lower part of the face. Results are even better than LGBP for the AUs 15, 17, 18, 25 and 26 (68.7% 67.1% 75.3% 69.6% and 67.5% against 63% 65.8% 70.4% 59.8% and 64.3% respectively). Results obtained for the upper part of the face are obviously not of a big importance (close to a random classifier's response) using mouth information. Only the AU 6 is well detected. This is because this AU (cheek raiser) often appears with the AU12 (the smile). With eyes AAM, results are just slightly better than random classifiers (56.8% where a random system does 50%). This can be explained by the difficulty in fitting AAM with enough accuracy to detect AUs. The eyebrows, for example, are difficult to localize, especially when hair hides them.

Regarding the fusion, we notice that the eyes AAM does not increase performances if coupled with LGBP histograms. But the mouth AAM or both AAMs coupled with LGBPs lead to the best results. Surprisingly, the detection of the upper part face AUs is improved with the mouth AAM: 81.8%, 83.4%, 80.9% 71.0% for the AUs 1 2 6 and 7 respectively against 78.8% 77.1% 77.0% and 68.5% with LGBP only. As previously mentioned, the improvement for the AU 6 can be explained by the fact that this AU is related to the AU 12. However the improvement brought by the fusion for the other AUs is more difficult to interpret. The multi-kernel classifier may use the information given by the mouth AAM not to directly detect these upper part AUs, but to have information about the subject (for example, information about its identity, skin type...) that can help the classifier to better analyze LGBP features and increase the precision of the AUs detection. This shows the interest of combining two types of different features.

Overall, the fusion of both AAMs with LGBP increases experimental results significantly for 9 over 12 AUs, the AUs 1 2 6 7 12 17 18 and 25. Results are worst only for one AU, the AU 15.

Finally, if we compare results in the person-specific and person-independent cases, we notice that the fusion is better than using only one feature type. In both cases, we get the highest 2AFC scores when combining LGBP with the mouth AAM: 71.6% for the person-specific case and 69.5% for the person-independent one). Combining with both AAMs gives approximately equal results to those obtained by combining with the mouth AAM: 71.5% for the person-specific case and 69.0% for the person-independent case.

4.1.3.2.2 Comparison with the other participants – To compare the participants' results in the AU sub-challenge, the F1-measure was used. The F1-measure considers both the precision p and the recall r of the test results to compute the score: p is the number of correct detections divided by the number of all returned detections and r is the number of correct detections divided by the number of detections that should have been returned. The F1-measure can be interpreted as a weighted average of the precision and recall, where an F1-measure reaches its best value at 1 and worst score at 0. The F1-measure is used only to optimize the part of the system converting signed values to binary values (size of the average filter and thresholds).

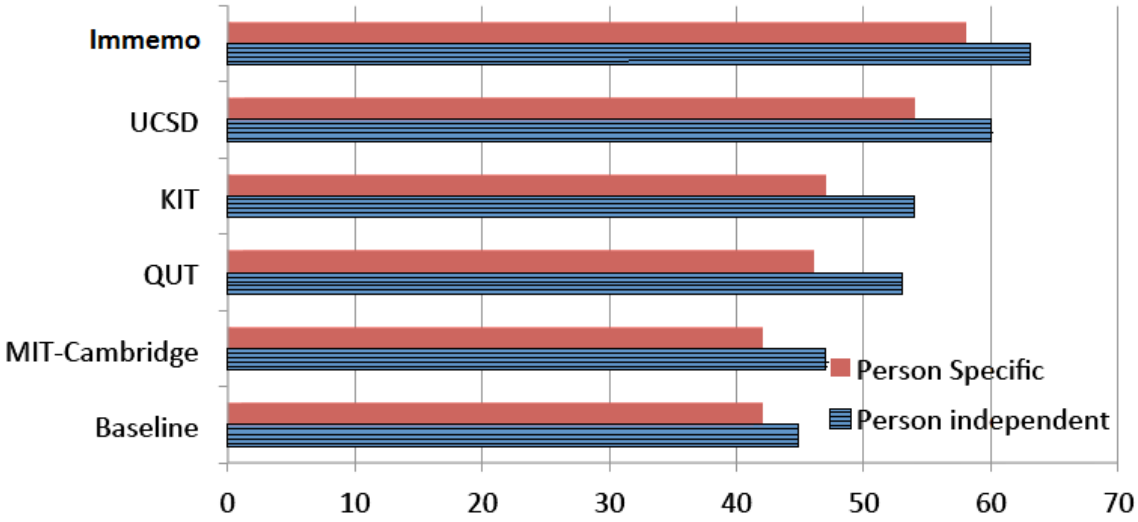


Figure 4.9: FERA AU sub-challenge official F1 results of all participants. UCSD: University of San Diego [WBR+11]. KIT: Karlsruhe Institute of Technology. QUT: Queensland University of Technology [CLL+11b]. MIT-Cambride : Massachusetts Institute of Technology and University of Cambridge [BMB+11].

$$F = 2 \cdot \frac{p \cdot r}{p + r} \tag{4.3}$$

The F1 scores obtained this way during the challenge and those of all participants are reported in Fig. 4.9. The system described in this thesis outperformed all other systems in the person independent and in the person specific case.

Classifiers were trained with the GEMEP-FERA, CK and Bosphorus databases for the AUs 1 2 4 12 15 17. We exclude the Bosphorous database for the training of the other AUs since these do not exist in this database.

If the reader is interested in the comparison between the different participants, he can refer to the paper of [SRS+11].

As a conclusion, we can state that by combining AAM features and LGBP features together with the use of temporal information, we succeed at overcoming the different approaches of the baseline and the other participants.

4.1.3.2.3 Emotion recognition results – Even though we have not participated in the emotion detection sub-challenge of FERA 2011, however, we have adapted our AU classifier to this task afterwards [SRS+12]. Table 4.2 reports the results of classification of the five emotions: Anger, fear, joy, relief and sadness. The overall results are compared to those of the emotion detection challenge, this is shown in the figure 4.10. As the figure

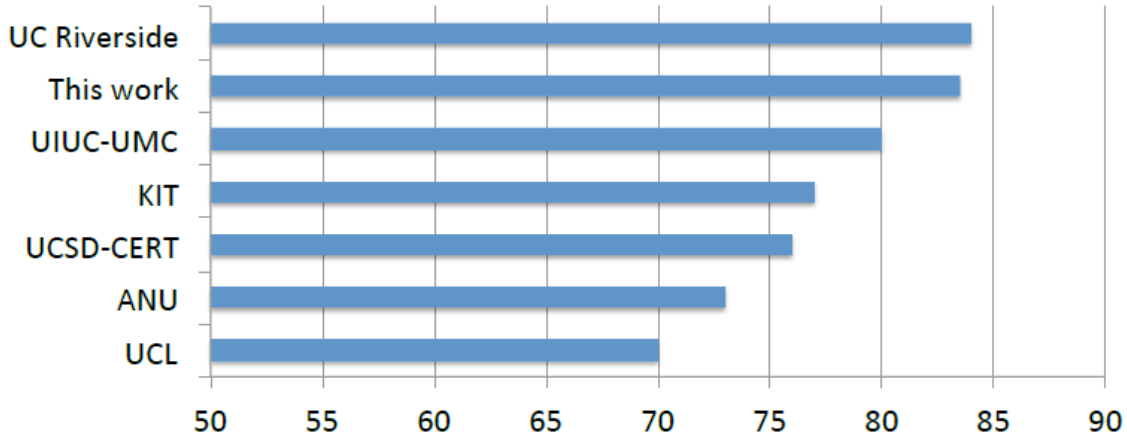


Figure 4.10: FERA emotion sub-challenge official F1 results of all participants. UCSD: University of San Diego [WBR+11]. KIT: Karlsruhe Institute of Technology. QUT: Queensland University of Technology [CLL+11b]. MIT-Cambride : Massachusetts Institute of Technology and University of Cambridge [BMB+11].

shows, our results come in the second place which confirms that our system is flexible to deal with a different kind of data.

	PI	PS	Overall
Anger	92.9	100	96.3
Fear	46.7	90	64
Joy	95	100	96.8
Relief	75	100	84.6
Sadness	60	100	76
Average	73.9	98	83.5

Table 4.2: Our team's emotion recognition classification rates on the testing database. We show performance for the Person-Independent (PI) case, Person-Specific (PS) and overall partition.

4.2 The Audio/Visual Emotion Challenge

AVEC 2012 [SVCP12] is the second International Audio-Visual Emotion recognition challenge. The goal of the challenge is to recognize four continuously valued affective dimensions [FSRE07]: arousal, valence, power, expectancy. Arousal is the dimension that indicates if the person is passive or active, valence shows if the person is pleasant or unpleasant, power is an indication of power versus weakness, and expectancy signifies novelty and unpredictability compared with expectedness or familiarity. It is constituted of two sub-challenges: The Fully Continuous Sub-Challenge and the Word-Level Sub-Challenge. We have participated in the Fully Continuous Sub-Challenge where the objective is to predict the values of the four dimensions at every moment during the recordings of a number of videos. This challenge is the first of its kind to call for the use of different modalities (audio, video, lexical and contextual) or their combination to detect emotion.

The challenge uses a part of the SEMAINE database [MVCP10] which presents naturalistic video and audio of human-agent interactions. It involves a user interacting with emotionally stereotyped characters. For the recordings, the participants are asked to talk in turn to four emotionally stereotyped characters. These characters are Prudence, who is even-tempered and sensible; Poppy, who is happy and outgoing; Spike, who is angry and confrontational; and Obadiah, who is sad and depressive.

Figure 4.11: Some examples of the SEMAINE database used in the context of AVEC 2012.

The difficulty of this challenge is primarily due to the nature and amount of data that should be dealt with. Actually, a big amount of unsegmented, non-prototypical and non preselected videos presenting subjects with naturalistic behavior is to be tested on. Figure 4.11 shows some images of the database. Three partitions of a part of the SEMAINE database are used: a training, development, and test partition, each consisting of 8 recordings of 8 different users. The training partition contains 31 sessions, while the development and test partitions contain 32 sessions. Test is done on subjects that can not be present in the training database.

12 participants from the communities of acoustic audio analysis, linguistic audio analysis and video analysis competed to win this challenge. Immemo has sent two teams on two different methodologies to participate in this challenge. These teams took the first and the second place. We were on the second team in collaboration with Dynamixyz and TelecomParisTech. We present a multi-modal system that extracts and merges visual, acoustic and context relevant features. Since our team works in video analysis, this

challenge was a chance for us to explore the audio and context domains and to employ our video analysis skills for the treatment of real data. Our major contribution was in the extraction of visual features in the form of laugh variations. This extraction is based on what we call the "Multi-Model AAM" (MM-AAM). In the following, we describe our global system (section 4.2.1). We detail our contribution at the level of facial features extraction using the MM-AAM (section 4.2.2).

4.2.1 Global system

The system takes as input a number of relevant features, fuses them using a fusion method and gives as output the four emotional dimensions (cf. figure 4.12).

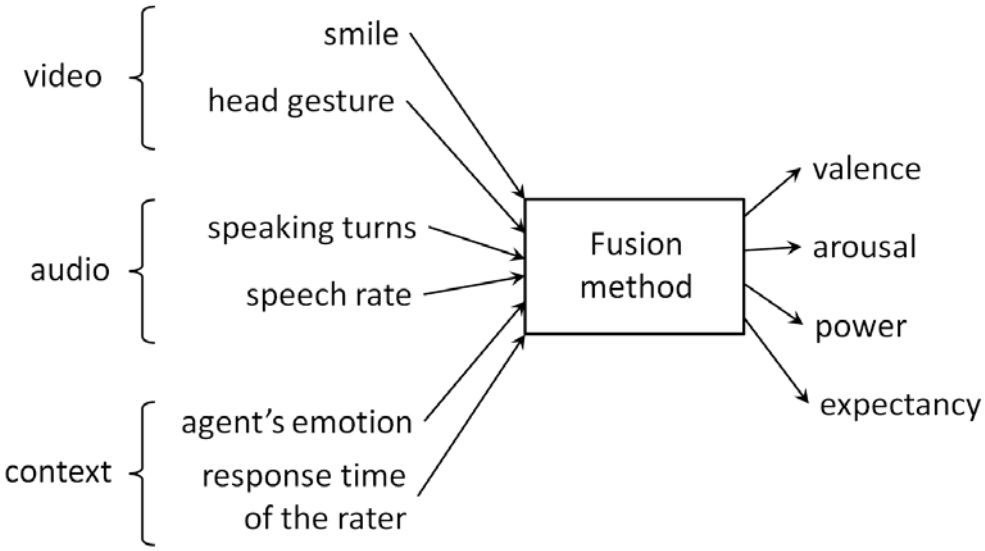


Figure 4.12: Overall view of the proposed method: a fuzzy inference system transforms the relevant features from video, audio and context into 4 emotional dimensions.

4.2.1.1 Relevant features extraction

To choose what features to extract and which are relevant, visual analysis of the videos and the ground truth emotional labels of the training and development databases (annotations of raters using FEELTRACE [CDCS+00]) was performed. By noticing which facial gestures and which audio characteristics where the most influencing on the ground truth labels of videos, we have revealed the most relevant features. The chosen features were those that explain the global trend of the emotion dimensions and not the small subtle variations of the emotions. They can be classified into:

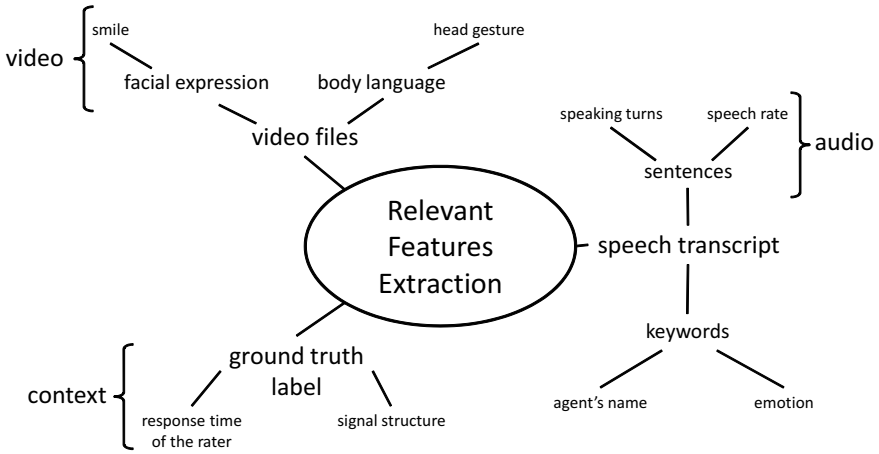


Figure 4.13: Sources of the relevant features: video files, speech transcripts and emotional labels.

- video features, that include facial expressions (especially laughter) and body language (especially head gesture);
- audio features, that include speaking turns and speech rate;
- context features, that include the emotional agent of the conversation (agent's name or emotional words that are said during the conversation), the response time of the rater and the conversation time.

The features extraction is made from 3 different data sources : videos, speech transcripts and ground truth labels (see figure 4.13).

Audio features – From speech transcripts, information about the speaking turns and the speech rate were extracted.

Speaking turns – The analysis of the sentences gives the length of the sentences pronounced by the subject. In our system, we use binary information. For each speaking turn in a conversation, if the number of words pronounced by the subject is high (above 35 words, empirical data learnt on training and development databases), the sentence is long ; otherwise, the sentence is short.

Speech rate – The speech rate is computed from the transcripts by the rate: number of words by time unit.

Context features Context features include information about the agent that is being spoken with (empathy), the response time of the rater and the signal structure.

Empathy – The conversations are performed between a subject and an emotional agent, which is set in one of the four quadrants of arousal-valence space (Spike is aggressive; Poppy is cheerful; Obadiah is gloomy; and Prudence is pragmatic).

In consequence, we remark the effect of the emotional agent on the emotion displayed by the subject. This can be translated by empathy. That is, the propagation of one's emotion to the other. For example, if the agent is Poppy, then the subject speaking to Poppy has a tendency to display behaviors of high valence and high arousal.

To find automatically which is the emotional agent of the sequences, we extract names from keywords detected from the speech transcripts. They provide some contextual information on which emotional agent the subject is speaking to.

Response time of the rater – The analysis of the ground truth labels highlights a delay in the start of annotations. This may be due to the initialization of the tool used to rate and to the response time of the rater, so that the first seconds of the ground truth labels may not be representative of the subject's emotion. Consequently, for the challenge, we modeled this behavior with a feature as a decreasing linear function on the first 20 seconds of the conversation.

Video features – Our main contribution involved the video features extraction. From the video files, body language and facial expressions were found to be the most relevant to the detection of the emotional dimensions.

Body language – We computed the global movement of the head pose in the scene. The video data are analyzed using a person-independent AAM [CET98b] built on the training and development databases. In the test phase, the pose parameters of the face are computed from the AAM model. The body movement is computed from the standard deviation of the head pose in a video sequence with a sliding temporal window of 40 seconds. The more the subject moves and makes wide movements, the higher this quantity is.

Laughter detection – In order to detect the facial expression, the system of [SSS12] was adopted. Actually, the work in this part of the system was divided into two. The first concerned the extraction of the video features and the second concerned the laughter detection. My principal contribution took place in the first part.

The overall process of laughter detection consists of four steps. First, facial features detection is performed using a Multi-Model person-independent AAM of which was my major contribution (section 4.2.2). Second, a person-specific appearance space is computed. Third, the appearance space is transformed into a person-independent expression space. Finally, expression recognition is performed and the laughter is deduced.

1. *Facial features detection by a person-independent Multi-Model AAM:* Section 4.2.2 is dedicated to this part. The facial features and appearance parameters of the MM-AAM are needed in various stages of the facial expression recognition system used in this challenge.
2. *Person-specific appearance space computation:* The neutral shape of each subject is computed, then 8 "plausible expressions" are added to this neutral. PCA is performed on the resulting shapes (neutral + expressions) to create a person-specific assumed shape model. This results in the shape parameters for each of these expres-

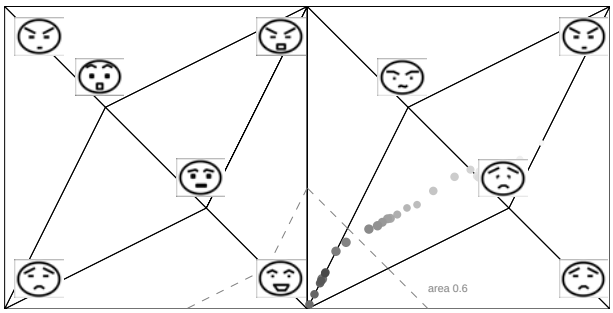


Figure 4.14: Trajectory of one subject's smile in the person-independent organized expression space

sions and the neutral. The computation of the neutral is done using the appearance parameters computed by the MM-AAM: A neutral of a subject is defined to be the face having the AAM parameters the closest from the mean AAM parameters of the images when the subject is not speaking.

3. *Appearance space into an expression space transformation:* In [SSS12], it was shown that the expression of a person can be defined by its relative position with respect to the other expressions and not by its absolute position in the appearance space. The organization of the expressions in the space with respect to each other was shown similar among different subjects. Using this invariant representation, we perform expression recognition.
4. *Expression recognition:* Now that the expression space is defined, an expression is recognized by defining an area in the manifold and computing the percentage of frames in this area. The direction of one expression is given by barycentric coordinates of the encompassing triangle and the intensity is between 0 (neutral) and 1 (high intensity). Figure 4.14 shows an extract of a video sequence that displays a smile in this space.
5. *Laughter deduction:* In our system, a smile is defined by a direction that is close to the expression E4 (corresponding to a coefficient above 0.6) and an intensity greater than 0.3. The feature "laughter" is defined by the percentage of images representing an expression of smile during a time window of 40 seconds.

4.2.1.2 Training Process

To find the source of the main variations of the 4 emotional dimensions, we computed the correlation between the ground truth labels of each sequence of development database and a signal that gives one of the relevant features described in section 4.2.1.1. We then compute the average value of these correlation coefficients. A high value of the mean value of the correlation coefficients indicates that the feature can be used to define the global shape of the variations of the emotional dimension.

The correlation analysis gave us the possibility to specify a set of rules. These rules can be summarized by:

- A high correlation between laughter and valence, which is normal, since laughing certainly means that the person is positive.
- A high correlation between laughter and arousal. Indeed, when subjects laugh, they are active.
- Body movement gives a good correlation on arousal. However, considering that the value is not high enough, we do not use it in our system.
- A high correlation is obtained between the structure of the speaking turns (long or short sentences) and expectancy. This high correlation is logical. Actually, when a subject is saying a long phrase, it is more likely that he is not surprised and thus the expectancy is low. On the other hand when he says a short sentences, he is probably responding to the emotional agent. Brief responses (short sentences) imply that the conversation is unexpected, thus a high expectancy.
- Speech rate is linked with power, but the correlation is low. This means that sometimes, when subjects speak fast, they are confident.
- The response time of the rater characterizes arousal and power with high correlation.
- A square-wave signal at the beginning of the conversation confirms the global change in expectancy during a conversation.

4.2.1.3 Fusion of relevant features

Fusion of the relevant features is done by two fusion systems: a Fuzzy Inference System (FIS) or a Radial Basis Function system (RBF). Both systems take as input the same relevant features that result from emotional states. The output is a continuous prediction of 4 emotional dimensions: valence, arousal, power and expectancy. For details about these fusion systems see [SSSS13].

4.2.2 Facial Features detection: The Multi-model AAM

As previously said, our main contribution concerned facial features extraction. This section details this contribution.

We have indicated in section 4.2.1.1 that the appearance parameters of the AAM are needed in several stages of the facial expression system used in this challenge. Precisely, they are needed to compute the neutral of the subjects and in the smile detector.

We thus, contribute in the computation of the neutral of every subject which results in the person-specific appearance space. After, this is transformed into an expression space. To find the expression of a subject in a specific frame, we find the appearance parameters of this frame of which are then projected to the expression space to find the intensity and the kind of the expression.

To arrive at a reliable expression detection, shape alignment using AAM should be as much accurate as possible especially at the level of the mouth because the smile was proven

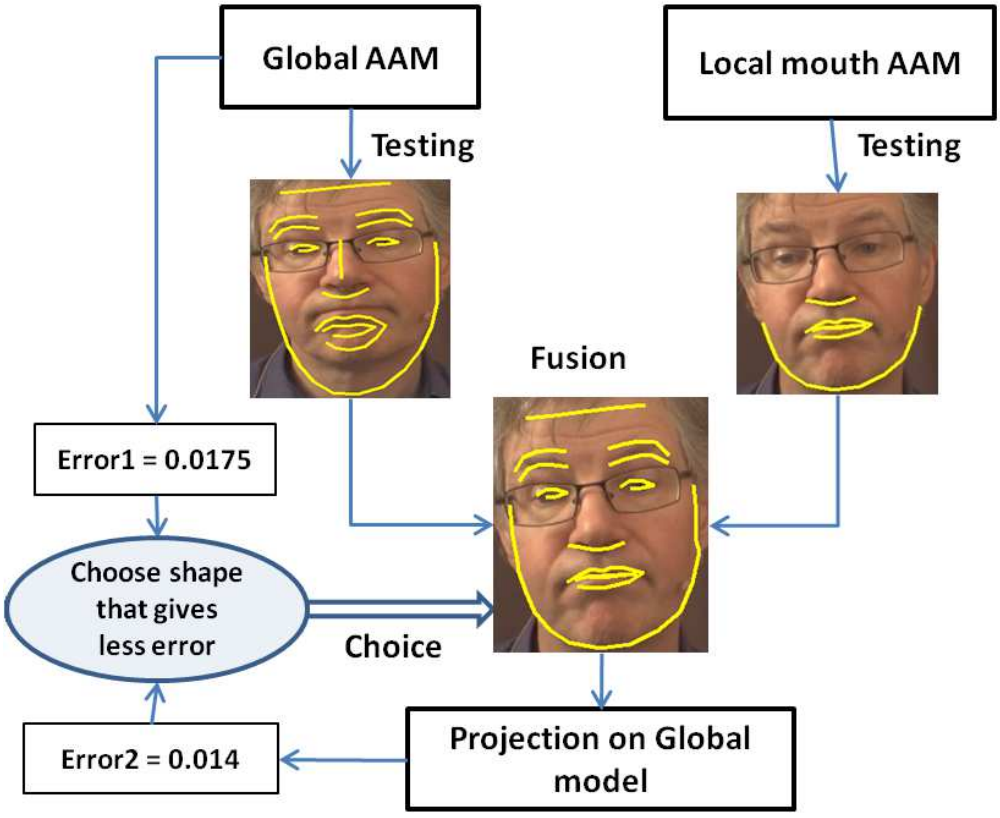


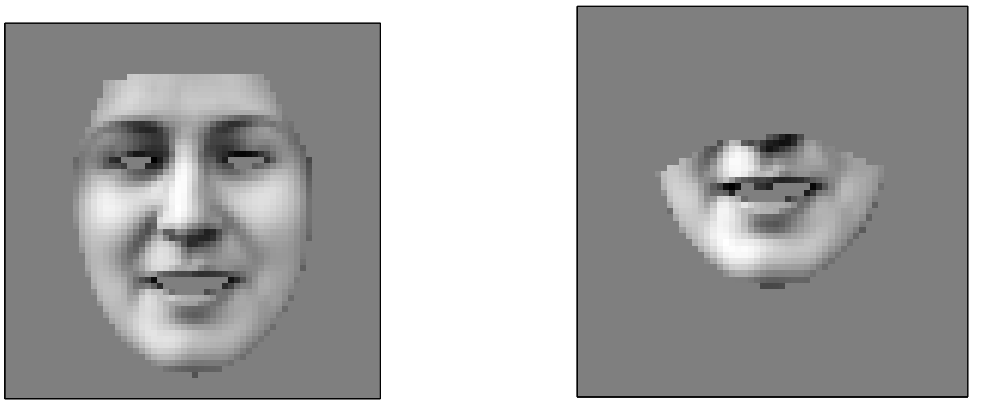
Figure 4.15: Example of person-independent Multi-Model AAM (MM-AAM)

to be a relevant feature for our emotion detection system.

We have signaled in the state-of-the-art approaches combining the advantages of global and local models (cf. section 2.3.1.4). We propose a scheme that belongs to this class of approaches. Our model combines extrinsically a global model of the face and a local model of the mouth. Only a local model of the mouth is integrated since mouth information is the most interesting for our system. In the following we detail our approach.

4.2.2.1 Proposition

Global models of the face have the advantage of making use of the correlations between the different features of the face to converge. Local models do not have this property. However, they can be more accurate locally (for example in the lower part of the face) than a global model when there are misleading variations on the upper part of the face such as hair, wicks or eyeglasses. As a consequence, a compromise solution should be implemented.



(a) GF-AAM mean texture (b) LM-AAM mean texture

Figure 4.16: Mean models of the GF-AAM and the LM-AAM

We propose the Multi-Model AAM (MM-AAM) (cf. figure 4.15). This MM-AAM combines the results of a Local Mouth AAM (LM-AAM) and a Global Face AAM (GF-AAM). These two models are trained on the same set of images of which are chosen from the train dataset of AVEC 2012. Holes are put in the place of the eyes and the mouth areas. Figure 4.16 shows the mean textures of these two models. The idea is to automatically choose the best shape between the GF-AAM and MM-AAM. This permits to take advantage of the precise localization of the mouth by LM-AAM when there is hair covering the face and the ability of the GF-AAM to generalize to new faces by using the correlations between the different parts of the face for the other cases.

Moreover, automatically choosing the most accurate shape among two or more shapes resulting from completely different models is not straightforward. As a matter of fact, to determine the best fit among two, the corresponding pixel errors are to be compared. Consequently, coming from different models, errors are not comparable. For this reason, a suitable scheme should be implemented. We thus propose to project the shape coming from the LM-AAM on the global model (GF-AAM) to obtain the parameters of GF-AAM that permit to give the same shape coming from LM-AAM. The pixel error is then calculated and compared to the shape coming from GF-AAM. The following describes in detail the steps of the algorithm.

Algorithm –

1. Train both models: GF-AAM and LM-AAM;
2. Apply both models on the testing videos: Get the global face shape S_{GF} and the local mouth shape S_{LM} ;
3. Substitute mouth shape from the LM-AAM in the shape from the GF-AAM: get the

Multi-Model shape S_{MM} ;

4. Project S_{MM} on the GF-AAM to obtain the corresponding appearance parameters and the projection error:
   (a) Align the S_{MM} to the mean shape of GF-AAM: ${S_{MM}}^{aligned}$;
   (b) Find the shape parameters b_s^{MM} corresponding to ${S_{MM}}^{aligned}$ using ${S_{MM}}^{aligned} = \overline{S_{GF}} + {V_s}^{GF}{b_s}^{MM}$. $\overline{S_{GF}}$ is the mean shape of the GF-AAM. ${V_s}^{GF}$ are the shape eigenvectors of the GF-AAM;
   (c) Warp the texture under S_{MM} into the mean shape of the GF-AAM $\overline{S_{GF}}$: this gives g_{MM};
   (d) Find the texture parameters ${b_g}^{MM}$ using $g_{MM} = \overline{g_{GF}} + {V_g}^{GF}b_{g_{MM}}$. $\overline{g_{GF}}$ is the mean texture of the GF-AAM. ${V_g}^{GF}$ are the texture eigenvectors of the GF-AAM;
   (e) Concatenate ${b_s}^{MM}$ and ${b_g}^{MM}$:
   $b_{MM} = \binom{W_s{b_s}^{MM}}{{b_g}^{MM}}$
   $W_s$is the weighting between pixel distances and intensities;
   (f) The projected appearance parameters are then: $C_{MM} = V_c b_{MM}$;
   (g) Synthesize the model texture ${g_{MM}}^{model}$ from the appearance parameters;
   (h) Compute the corresponding projection error: $E_{MM} = \left|{g_{MM}}^{model} - g_{MM}\right|$;
5. Choose the shape (S_{MM} or S_{GF}) that gives the lowest projection error;

Figure 4.17: An example of an image where neither the GF-AAM nor the LM-AAM succeed to converge because the person goes out from the frame of the camera

Confidence extraction

Our Multi-Model scheme has proven to efficiently alternate between the local and the global models (cf. section 4.2.2.2). Nevertheless, in cases where both models fail, due to very noisy information such as when a person face comes out from the image (cf. figure 4.17), a scheme should be implemented to exclude such cases so that they wont introduce noise to the global system. We propose to assign each frame a binary confidence that indicates if the appearance information of the frame should be took into account in the system or not. This confidence is computed based on the analysis of projection errors of the sequence in question. As a matter of fact, a threshold error is set for every sequence. If the error of one frame is less than or equal to this threshold error, then the frame is considered to have a good alignment and thus is given a confidence index of 1, else it is assigned a confidence index of 0.

The threshold error is obtained through a simple scheme: For a sequence, find the mean and maximum of the optimal projection errors: E_{mean} and E_{max} respectively.

$$E_{threshold} = \begin{cases} 1.1 \times E_{mean} \text{ if } E_{max} >= a \\ E_{max} \text{ if } E_{max} <= b \\ 1.2 \times E_{mean} \text{ else} \end{cases}$$

The values of a and b are set to 0.02 and 0.017 respectively. These values are set this way due to our observation of the errors of the train and development databases. Actually, we have remarked that in all the sequences, when there is divergence or really bad alignment, the error was in the order of 0.2.

4.2.2.2 Results of the MM-AAM

In order to compare the performance of the proposed Multi-Model AAM to that of the Global AAM, we plot the Ground Truth Error (GTE) versus the percentage of aligned images for one sequence of the test database. The GTE is the mean of the Euclidean between ground truth (real locations of eyes centers, mouth center and the nose tip) marked manually and the points given by the shape extraction method normalized by the distance between the eyes. The subject in this sequence is smiling most of the time with a smile of varying intensity. Thus, the comparison on such a sequence is significant since our system uses a smile detector to detect the emotional dimensions and consequently this smile detector uses AAM results. The GTE of both the MM-AAM and the Global AAM are shown in figure 4.18. The figure shows that with a GTE less than 10% of the distance between the eyes, the MM-AAM is able to extract facial features of 96% of the total images of the sequence, compared to 82% by the Global AAM. Actually for this sequence the local mouth model performs better than global face model at the level of the mouth. So, the MM-AAM chooses the combination of both. Figure 4.19 shows qualitative results on some images of this sequence. This figure shows three cases, in the first case, the subject is smiling wide, in the second, he smiles a small smile after a wide one and in the third, he opens his mouth while speaking. As we see, in the first case, the global model fails to give precise results at the level of the mouth because of the wide smile.

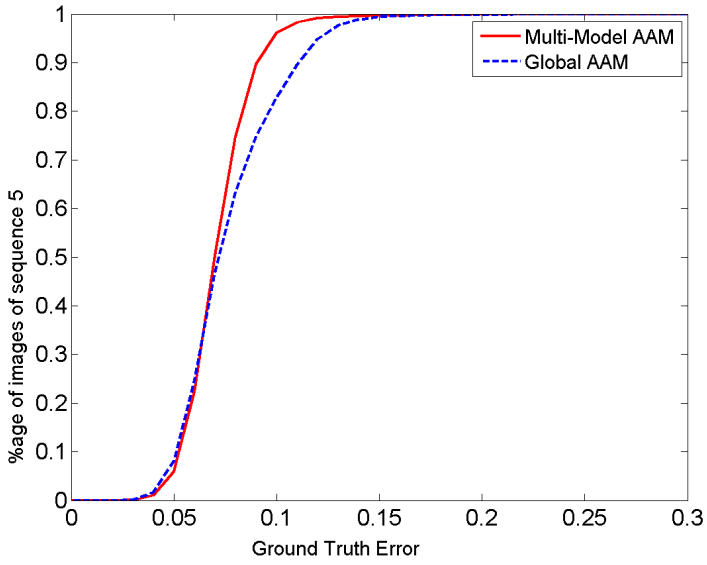


Figure 4.18: Comparison between the GTE of the Multi-Model AAM and the Global AAM for one sequence of the tests database.

However the MM-AAM gives the precise result because of its local mouth model. In the second case, the GF-AAM fails because the AAM parameters are initialized by those of the preceding image which is a wide smiling one. In the third, the small contrast between the teeth and the skin makes a global model fails while a local one does not. Figure 4.20 shows the results of both the GF-AAM and the combination of the GF-AAM and the LM-AAM for another sequence of the test database. In the case of this sequence, the local mouth model performs poorer than the global model. The reason is that the subject has a beard and the local model was not trained on such subjects. The global model makes use of the relationships between the upper part and the lower one to converge even if the training database does not contain such subjects. Thus the MM-AAM chooses the results of the GF-AAM rather than the combination of both for most of the frames of the sequence. As a conclusion, employing the MM-AAM is efficient in alternating between results of a global AAM and a local one according to the one that performs better which permits to take advantage of both global and local frameworks.

4.2.3 Emotion detection results

Table 4.3 shows the results of both fusion systems on the test database of the challenge. This is expressed as the correlation between the system's results and the mean of raters ground truth evaluation. The learning has been performed on training and development

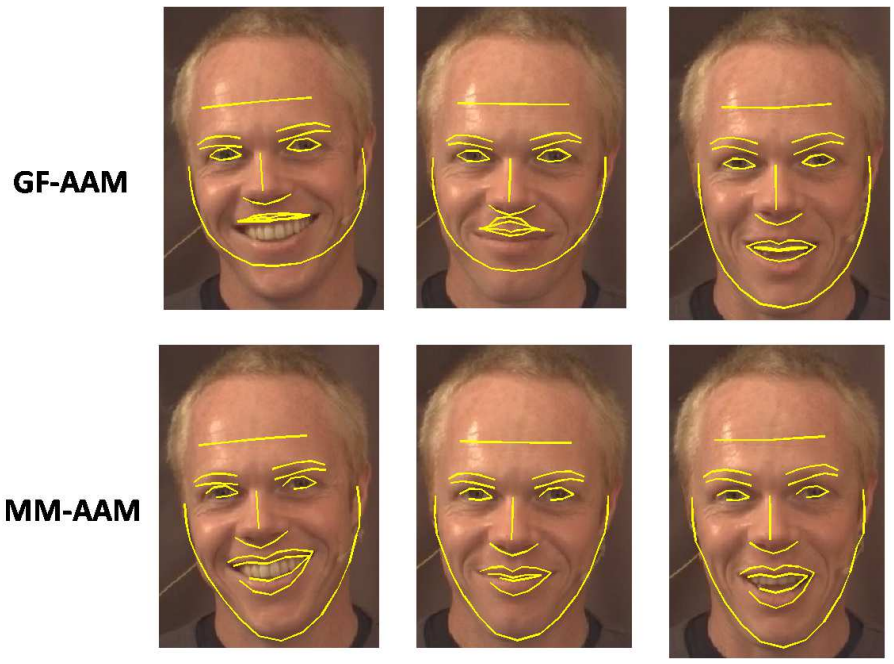


Figure 4.19: Comparison between the GF-AAM (top row) and the MM-AAM (bottom row) on one sequence of the test database

databases. We compare our results to those of the winners [NRB+12] and the team who came in the third place [SCS+12].

First, comparing the results of our system using two different fusion methods, we obtain the same mean correlation (0.43), this shows that our results are stable whatever the fusion system used. This means that both methods generalize correctly. Moreover, we achieve similar performance to the winners of the challenge (0.46). Concerning the other challengers, those were further behind (0.34 for the third competitor).

In addition, we have computed the mean correlation coefficient between one rater and the other ones (last row of the table) and compared our system to it. Actually, four different raters have labeled the four emotional dimensions on each video sequence. We were interested in figuring out how much these are in agreement between each other. Surprisingly, we have obtained a low correlation between the different labels of these raters. This means that it is hard for two humans to agree on the emotional state of a subject. Moreover, comparing our system's results to the correlation between the annotators, we notice that they are quite similar: we have a mean correlation of 0.43 vs. 0.45 for the annotators. This result shows that our automatic system performs as much as good as a human annotator and thus can replace it.

Figure 4.21 demonstrates the position of our team with respect to the position of the

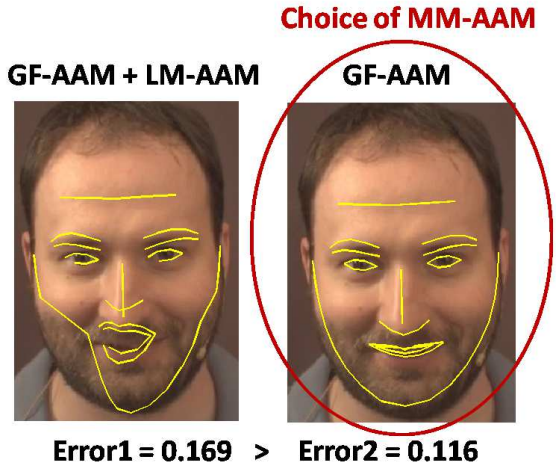


Figure 4.20: Example of the MM-AAM in the case where the algorithm chooses the GF-AAM rather than the combination of the GF-AAM and the LM-AAM

other teams in the challenge.

4.3 Conclusion

In this chapter we have described how we have adapted our skills in facial features detection using Active Appearance Models in the context of two grand challenges: the Facial Expression Recognition and Analysis Challenge (FERA 2011) and the the Audio/Visual Emotion Challenge (AVEC 2012). Facial features extraction using AAM constitute the basic component of both systems. In FERA 2011 the appearance parameters of AAM

Methods Challenge's Position	[NRB+12] 1^{st}	FIS 2^{nd}	RBF 2^{nd}	[SCS+12] 3^{rd}	Raters
Arousal	.61	.42	.42	.36	.44
Valence	.34	.42	.42	.22	.53
Power	.56	.57	.57	.48	.51
Expectancy	.31	.33	.32	.33	.33
Mean	.46	.43	.43	.34	.45

Table 4.3: Results comparing our system to the winner of the challenge, the participant that took the third place, and the mean correlation between one rater and the other ones. Results comparing our two fusion systems are presented as well.

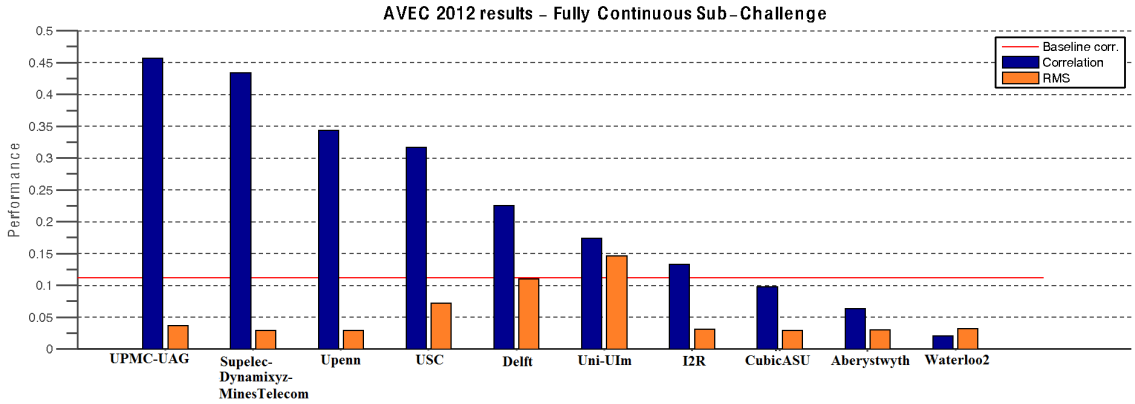


Figure 4.21: Position of our team (Supelec-Dynamixyz-MinesTelecom) with respect to the position of the other teams in the AVEC 2012 challenge

constitute the geometric feature component of an Action Unit detector that combines geometric and appearance features via a multiple kernels approach. In AVEC 2012, appearance parameters are used in the expression recognition system used by an emotion detection system.

Our contributions concern the extraction of facial features from the face. First we have explored local and global models for the FERA challenge. In the AVEC challenge, we have proposed a Multi-Model AAM that combines a global model of the face and a local model of the mouth. The model efficiently switches between these two models by comparison of projection errors on the same global model.

We like to point out that our initial purpose from the participation in the AVEC 2012 was to apply our gaze detection model proposed in the previous chapter in the context of multi-modal emotion detection. This is due to the fact that gaze plays a major role in perceiving emotion [LTP08]. However, after the analysis of data during the design of the system (cf. section 4.2) and due to the large amount of videos to deal with, this turned to be an out-of-scope task.

Conclusion

Summary

The work in this thesis dealt with the automatic detection of non-verbal cues of human beings during interaction with the computer. Among these cues we have concentrated on eye gaze, blink, expression and multi-modal affect recognition.

In the first part, we have proposed the Multi-Object Facial Actions Active Appearance Model (MOFA-AAM). The model combines statistical modeling of the face and parameter-based models in the context of a multi-objective optimization. The specificity of the proposed model is that different parts of the face are treated as separate objects and eye movements are extrinsically parameterized (movement of the iris and eyelids). The parameters are interpretable and can be used in important applications such as the detection of gaze and blinking. From a learning database that contains no variations in gaze and blinking (the people in frontal view and they watch in front of them) the model is able to follow the movement of the iris and the eyelids, which increases the robustness of active appearance models (AAM) by restricting the amount of variation in the learning base. The multi-objective framework makes the model more robust to head pose. Specific parts of the face are favored over the others in function of the head pose.

The second part of the thesis concerns the use of face modeling in the context of expression and emotion recognition. First we have proposed a system for the recognition of facial expressions in the form of Action Units (AU). The proposed system is based on the combination of Local Gabor Binary Pattern Histograms (appearance features) and AAMs (hybrid features: appearance and geometry) using Multi-Kernel Support Machine Vectors. Our contribution concerned mainly the extraction of AAM features. The AUs to detect concerned the upper and lower part of the face. Thus, we have opted for the use of local models to extract these features. Results have demonstrated that the combination of AAM with the LGBP appearance features has led to ameliorate the results of recognition. This system was evaluated in FERA 2011, an international challenge for emotion recognition of which our team have took the first place.

The second system concerns the multi-modal recognition of four continuously valued affective dimensions: arousal, valence, power and expectancy. We have proposed a system that fuses audio, context and visual features and gives as output the four emotional dimensions. The visual features are in the form of facial expressions. More precisely, we have

found that the smile is a relevant cue for the detection of the aforementioned dimensions. To detect this feature, AAM is used to delineate the face. We contribute at this stage of the system to find the precise localization of the facial features. Accordingly, we propose the Multi-Local AAM. This model combines extrinsically a global model of the face and a local one of the mouth through the computation of projection errors on the same global AAM. The proposed system was evaluated in the context of AVEC 2012 challenge and our team got the second place with very close results to those who came in the first place.

Perspectives

The work of this thesis has done place to new research axis.

Aggregation of local and global models

We have seen in the work for FERA 2011 and AVEC 2012 how local appearance models are sometimes more accurate for modeling local regions of the face than global models. However, concerning the head pose, a global model is more robust.

We argue that optimizing local appearance parameters for the upper and lower parts of the face while one pose vector for the whole face would result in a better localization of the facial landmarks. Using regression as in the classical AAMs, the prediction of the update in pose parameters using the local models combined with the prediction coming from the global one may result in a better prediction of these parameters.

A single face model

Another continuity of the proposed model is to integrate all of the proposed parameters (gaze, blink, etc.) into a single face model. Instead of sequentially applying the eye skin model and the iris model, a more simple model would be to optimize the parameters of both models at the same time.

Mouth object

Lips deformation – As the concentration of this thesis was on the eyes region, a natural continuity is to model the mouth region. The objective is to reach an Active Appearance Model that is capable of analyzing the movements of the mouth without the necessity of including such variations in the learning database of AAM. Modeling the mouth region is a more complex task that of the eyes. Following the local landmark analysis that we have presented in chapter 3, we can extract a number of parameters that are responsible for the mouth movements.

Tongue and teeth – Same as we have modeled the eyeballs as separate objects from the facial skin, the tongue and teeth can also be modeled separately for the integration in active appearance models. In this way, as the lips deform, combined with the separate lips

and teeth models will generate valid mouth appearances without the necessity of including the interior of the mouth while learning the active appearance model.

Multi-Modal emotion recognition

Features extraction in our team's multi-modal emotion recognition system was based on our observation of the ground truth labels of the emotional dimensions during each video. From these ground truth labels we tried to visually find reliable information that affect the labels of emotions. During our observations we have noticed that eyes motions (gaze and blinking) have a correlation with the emotional dimensions.

In fact, several studies have pointed out the importance of gaze for multi-modal emotion recognition [SCGH05, ZPRH09]. [LM08] conducted an empirical study that explores how would an observer attribute the emotional state based on eye gaze. Moreover, [OSM12] used gaze information in their multi-modal affect recognition system.

We think that using gaze and blink information in our proposed emotion recognition system would increase effectively the recognition results. In addition, exploring relationships between eye movements and emotion would be a piste for future work. This can be done with the aid of the database of [SLPP12]. The authors present a new multimodal database that features simultaneous recordings of a set of participants faces using multi-cameras, speech, eye gaze, pupil size, peripheral/central nervous system physiological signals (respiration amplitude and skin temperature), together with their rating of their emotional feelings.

RGB-Z landmarks bio-inspired artificial vision. Application on the detection of the gaze and gestures

The work in this thesis has also done place for a new thesis in gaze detection. Rather than detecting gaze using deformable models which are conventional techniques as it is the case in this thesis, a bio-inspired approach to artificial perception is proposed. The objective is to produce a system that will learn independently to identify visual features (motion, texture, curvature of surfaces, normal etc.) as certain gestures such as the gaze.

Project MILES (FUI 2013): Multi-platform Interactive Learning with Experiential Systems

MILES is an online platform for immersive training where trainees and trainers find themselves in virtual classrooms. The content adapts in function of Media Connection: Smartphone, tablet, or PC simulator. The work about gaze detection in this thesis will be a starting point of the implementation of gaze detection in this project.

List of publications

The work in this thesis has done place for a total of 9 publications. We put in italics the publications that are the issue of a collaborative work.

Journals

1. **Salam, H.** and Seguier, R. and Stoiber N. (2013). Integrating head pose to a 3D Multi-Texture approach for gaze detection. International Journal of Multimedia & Applications. Vol.5, No.4, August 2013.
2. *Soladie, C. and **Salam, H.** and Stoiber, N. and Seguier R. (2013). Continuous Facial Expression Representation for Multimodal Emotion Detection. International Journal of Advanced Computer Science. Vol.5, No.4, 2013.*
3. *Sénéchal, T. and Rapp, V. and **Salam, H.** and Seguier, R. and Bailly, K. and Prevost, L. (2011). Facial Action Recognition Combining Heterogeneous Features via Multi-Kernel Learning. IEEE transactions on Systems Man and Cybernetics Part B (TSMC-B). Special issue on the facial recognition challenge 2011.*

International conferences

1. **Salam, H.** (2013). A 3D-Eyeball/Skin Decorrelated Active Appearance Model. In the 1st IEEE/IIAE International Conference on Intelligent Systems and Image Processing 2013.
2. **Salam H.**, Stoiber N., Seguier R. (2012). A Multi-Texture Approach For Estimating Iris Positions in the Eye Using 2.5D Active Appearance Models. In Proceedings of the IEEE International Conference of Image Processing 2012 (ICIP).
3. *Soladie C., **Salam H.**, Pelachaud C., Stoiber N., Seguier R. (2012). A Multimodal Fuzzy Inference System Using a Continuous Facial Expression Representation for Emotion Detection. In Proceedings of the 14th ACM international conference on Multimodal interaction - ICMI '12, United States.*
4. *Sénéchal, T. and Rapp, V. and **Salam, H.** and Seguier, R. and Bailly, K. and Prevost, L. (2011). Combining LGBP Histograms with AAM coefficients in the Multi-Kernel SVM framework to detect Facial Action Units. Proc. FG'11, Facial Expression Recognition and Analysis Challenge (FERA'11).*

National conferences

1. **Salam, H.** and Seguier, R. and Stoiber, N. (2013). Détection de l'iris dans des visages de pose quelconque : approche multi-textures et Modèles Actifs d'Apparence 2.5D. In Proceedings of GRETSI 2013, Groupe d'Etudes du Traitement du Signal et des Images.
2. *Sénéchal, T. and Rapp, V. and **Salam, H.** and Seguier, R. and Bailly, K. and Prevost, L. (2012). Combinaison de Descripteurs Hétérogènes pour la Reconnaissance de Micro-Mouvements Faciaux. In Proceedings of RIFA 2012, Reconnaissance des Formes et l'Intelligence Artificielle.*

Glossary

	A
AAM	Active Appearance Model
AMA	Abstract Muscle Action
AU	Action Unit
ASM	Active Shape Models
AUV	Action Unit Vector
AP	Animation Parameters
2AFC	2-Alternative Forced Choice task
AOM	Active Orientation Models
AVEC	Audio/Visual Emotion Challenge
	B
BTSM	Bayes Tangent Shape Modes
BAAM	Bilinear AAM
	C
CLM	Constrained Local Models
CK	Cohn Kanade
	D
DE-AAM	Double Eyes AAM
	E
EASM	Extended ASM
EM	Expectation Maximization
	F
FAP	Facial Animation Parameter
FAU	Facial Action Unit
FFD	Free-Form Deformation
FACS	Facial Action Coding System
FFP	Facial Feature Point
FAPU	Facial Animation Parameter Unit
FEM	Finite Element Models
FERA	Facial Expression Recognition and Analysis Challenge
	G

GAC	Geodisic active contours
GF-AAM	Global Face AAM
GA	Genetic Algorithm
GD	Gradient Descent
GTE	Ground Truth Error
GERG	Geneva Emotion Research Group
	H
HHI	Human-Human Interaction
HCI	Human-Computer Interaction
HOG	Histogram of Oriented Gradients
	I
ICA	Independent Component Analysis
ICP	Iterative Closest Point
IR	InfraRed
IMMEMO	IMMersion 3D basée sur l'interaction *È*MOtionnelle
	L
LM	Levenberg-Marquardt
LGBP	Local Gabor Binary Pattern
LBP	Local Binary Pattern
LM-AAM	Local Mouth AAM
	M
MRF	Markov Random Field
MM-AAM	Multi-Model AAM
MTAAM	Multi-Texture AAM
MOAAM	Multi-Objective AAM
MM	Morphable Models
MDL	Minimum Description Length
MPA	Minimal Perception Action
	N
NVP	Normal Vector Profile
NN	Nearest Neighbor
	P
PASM	Partial ASM
PCA	Principal Component Analysis
PFS	Pupil Feature Space
PG	Pose Gaze database
	R
RLMS	Regularized Landmark Mean-Shift
RBF	Radial Basis Function
	S
SUV	Shape Unit Vector

SP	Shape Parameters
SDP	Signed Distance Potential
SVM	Support Vector Machine
SOAAM	Single-Objective AAM

U

UImHPG	UIm Head Pose and Gaze database

Bibliography

[AD05] B. Abboud and F. Davoine. Bilinear factorisation for facial expression analysis and synthesis. *Vision, Image and Signal Processing, IEE Proceedings-*, 152(3):327–333, 2005.

[Ahl01] J. Ahlberg. Candide-3–an updated parameterized face. Technical report, Report No. LiTH-ISY, Dept. of Electrical Engineering, Linkoping University, Sweden, 2001.

[Ahl02] J. Ahlberg. An active model for facial feature tracking. *EURASIP Journal on Applied Signal Processing*, 2002(1):566–571, 2002.

[ARARCE11] S.E. Ayala-Raggi, L. Altamirano-Robles, and J. Cruz-Enriquez. Automatic face interpretation using fast 3¡ i¿ d¡/i¿ illumination-based¡ i¿ aam¡/i¿ models. *Computer Vision and Image Understanding*, 115(2):194–210, 2011.

[AS09] C.D.N. Ayudhya and T. Srinark. A method for real-time eye blink detection and its application. In *6th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, 2009.

[ASKK09] S. Asteriadis, D. Soufleros, K. Karpouzis, and S. Kollias. A natural head pose and eye gaze dataset. In *Proceedings of the International Workshop on Affective-Aware Virtual Agents and Social Robots*, page 1, 2009.

[Bac09] I. Bacivarov. *Advances in the Modelling of Facial Sub-Regions and Facial Expressions using Active Appearance Techniques.* PhD thesis, National University of Ireland, College of Engineering and Informatics, 2009.

[Bai10] K. Bailly. *Méthodes d'apprentissage pour l'estimation de la pose de la tête dans des images monoculaires.* PhD thesis, Université Pierre et Marie Curie-Paris VI, 2010.

[BCR+07] G. Bailly, A. Casari, S. Raidt, et al. Towards eyegaze-aware analysis and synthesis of audiovisual speech. In *Proceedings, International Conference on Auditory-Visual Speech Processing, AVSP 2007*, pages 50–56, 2007.

[BCZ93] A. Blake, R. Curwen, and A. Zisserman. Affine-invariant contour tracking with automatic control of spatiotemporal scale. In *Computer Vi-*

sion, 1993. Proceedings., Fourth International Conference on, pages 66–75. IEEE, 1993.

[BH05] A.U. Batur and M.H. Hayes. Adaptive active appearance models. *Image Processing, IEEE Transactions on*, 14(11):1707–1721, 2005.

[BI98] A. Blake and M. Isard. *Active Contours: The Application of Techniques from Graphics,Vision,Control Theory and Statistics to Visual Tracking of Shapes in Motion*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1st edition, 1998.

[BIC08] I. Bacivarov, M. Ionita, and P. Corcoran. Statistical models of appearance for eye tracking and eye-blink detection and measurement. *Consumer Electronics, IEEE Transactions on*, 54(3):1312–1320, 2008.

[BLF+06] M.S. Bartlett, G.C. Littlewort, M.G. Frank, C. Lainscsek, I.R. Fasel, and J.R. Movellan. Automatic recognition of facial actions in spontaneous expressions. *Journal of Multimedia*, 1(6):22–35, 2006.

[BMB+11] T. Baltrusaitis, D. McDuff, N. Banda, M. Mahmoud, R. el Kaliouby, P. Robinson, and R. Picard. Real-time inference of mental states from facial expressions and upper body gestures. In *Automatic Face & Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on*, pages 909–914. IEEE, 2011.

[BP07] I. Baran and J. Popovic. Automatic rigging and animation of 3d characters. *ACM Transactions on Graphics (TOG)*, 26(3):72, 2007.

[BQ06] Z. Baizhen and R. Qiuqi. Facial feature extraction using improved deformable templates. In *Signal Processing, 2006 8th International Conference on*, volume 4. IEEE, 2006.

[BS07] T. Banziger and K.R. Scherer. Using actor portrayals to systematically study multimodal emotion expression: The gemep corpus, 2007.

[BV99] V. Blanz and T. Vetter. A morphable model for the synthesis of 3d faces. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 187–194, 1999.

[CBB02] R. Cesar, E. Bengoetxea, and I. Bloch. Inexact graph matching using stochastic optimization techniques for facial feature recognition. In *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, volume 2, pages 465–468. IEEE, 2002.

[CC04] D. Cristinacce and T.F. Cootes. A comparison of shape constrained facial feature detectors. In *Automatic Face and Gesture Recognition, 2004. Proceedings. Sixth IEEE International Conference on*, pages 375–380. IEEE, 2004.

[CC06a] D. Cristinacce and T. Cootes. Feature detection and tracking with constrained local models. In *Proc. British Machine Vision Conference*, volume 3, pages 929–938, 2006.

[CC06b] D. Cristinacce and T.F. Cootes. Facial feature detection and tracking with automatic template selection. In *Automatic Face and Gesture Recognition, 2006. FGR 2006. 7th International Conference on*, pages 429–434. IEEE, 2006.

[CC07] D. Cristinacce and T. Cootes. Boosted regression active shape models. In *Proc. British Machine Vision Conference*, volume 2, pages 880–889, 2007.

[CCD00] A. Colburn, M.F. Cohen, and S. Drucker. The role of eye gaze in avatar mediated conversational interfaces. *Microsoft Research Report*, 81:2000, 2000.

[CCTG92] T.F. Cootes, D.H. Cooper, C.J. Taylor, and J. Graham. Trainable method of parametric shape description. *Image and Vision Computing*, 10(5):289–294, 1992.

[CDCS+00] R. Cowie, E. Douglas-Cowie, S. Savvidou, E. McMahon, M. Sawey, and M. Schroder. FEELTRACE: an instrument for recording perceived emotion in real time. In *Proceedings of the ISCA Workshop on Speech and Emotion*, 2000.

[CET98a] T.F. Cootes, G. Edwards, and C.J. Taylor. A comparative evaluation of active appearance model algorithms. In *British Machine Vision Conference*, volume 2, pages 680–689, 1998.

[CET98b] T.F. Cootes, G.J. Edwards, and C.J. Taylor. Active appearance models. In *IEEE European Conference on Computer Vision (ECCV '98)*, page 484, 1998.

[CET98c] T.F. Cootes, G.J. Edwards, and C.J. Taylor. Active appearance models. *Proc. IEEE European Conference on Computer Vision (ECCV '98)*, page 484, 1998.

[CK04] U. Canzler and K.F. Kraiss. Person-adaptive facial feature analysis for an advanced wheelchair user-interface. In *Conference on Mechatronics & Robotics*, volume 3, pages 871–876, 2004.

[CKLO09] Qiu Chen, Koji Kotani, Feifei Lee, and Tadahiro Ohmi. An accurate eye detection method using elliptical separability filter and combined features. *IJCSNS International Journal of Computer Science and Network Security*, 9(8), August 2009.

[CKS97] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. *International journal of computer vision*, 22(1):61–79, 1997.

[CLL+11a] S.W. Chew, P. Lucey, S. Lucey, J. Saragih, J.F. Cohn, and S. Sridharan. Person-independent facial expression detection using constrained local models. In *Automatic Face & Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on*, pages 915–920. IEEE, 2011.

[CLL+11b] S.W. Chew, P. Lucey, S. Lucey, J. Saragih, J.F. Cohn, and S. Sridharan. Person-independent facial expression detection using constrained local models. In *Automatic Face & Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on*, pages 915–920. IEEE, 2011.

[CPP08] M.D. Cordea, E.M. Petriu, and D.C. Petriu. Three-dimensional head tracking and facial expression recovery using an anthropometric muscle-based active appearance model. *Instrumentation and Measurement, IEEE Transactions on*, 57(8):1578–1588, 2008.

[CT99] T.F. Cootes and C.J Taylor. A mixture model for representing shape variation. *Image and Vision Computing*, 17(8):567–573, 1999.

[CT04] T.F. Cootes and C.J. Taylor. Statistical models of appearance for computer vision. Technical report, Imaging Science and Biomedical Engineering, University of Manchester,, 2004.

[CT06] T.F. Cootes and C.J. Taylor. An algorithm for tuning an active appearance model to new data. In *Proc. British Machine Vision Conference*, volume 3, pages 919–928. Citeseer, 2006.

[CT07] F.J.S Carvalho and J. Tavares. Eye detection using a deformable template in static images. In *VIPimage-I ECCOMAS Thematic Conference on Computational Vision and Medical Image Processing*, pages 209–215, 2007.

[CTC10] A. Caunce, C. Taylor, and T. Cootes. Adding facial actions into 3d model search to analyse behaviour in an unconstrained environment. In *Advances in Visual Computing*, pages 132–142. Springer, 2010.

[CTC12] A. Caunce, C. Taylor, and T. Cootes. Using detailed independent 3d sub-models to improve facial feature localisation and pose estimation. In *Advances in Visual Computing*, pages 398–408. Springer, 2012.

[CTCG95] T.F. Cootes, C.J. Taylor, D.H. Cooper, and J. Graham. Active shape models-their training and application. *Computer vision and image understanding*, 61(1):38–59, 1995.

[CV01] T.F. Chan and L.A. Vese. Active contours without edges. *IEEE Transactions on Image Processing*, 10(2):266–277, 2001.

[DB08] M. Divjak and H. Bischof. Real-time video-based eye blink analysis for detection of low blink-rate during computer use. In *First International Workshop on Tracking Humans for the Evaluation of their Motion in Image Sequences (THEMIS 2008)*, pages 99–107, 2008.

[DD04] F. Dornaika and F. Davoine. Head and facial animation tracking using appearance-adaptive models and particle filters. In *Computer Vision and Pattern Recognition Workshop, 2004. CVPRW'04. Conference on*, pages 153–153, 2004.

[DD06] F. Dornaika and F. Davoine. On appearance based face and facial action tracking. *Circuits and Systems for Video Technology, IEEE Transactions on*, 16(9):1107–1124, 2006.

[DMTP04] M. Dobes, L. Machala, P. Tichavskı, and J. Pospisil. Human eye iris recognition using the mutual information. *Optik-International Journal for Light and Electron Optics*, 115(9):399–404, 2004.

[DOG06] F. Dornaika, J. Orozco, and J. Gonzàlez. Combined head, lips, eyebrows, and eyelids tracking using adaptive appearance models. In *Articulated Motion and Deformable Objects*, pages 110–119. Springer, 2006.

[DRL+06] R. Donner, M. Reiter, G. Langs, P. Peloschek, and H. Bischof. Fast active appearance model search using canonical correlation analysis. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(10):1690–1694, 2006.

[E+93] Paul Ekman et al. Facial expression and emotion. *American Psychologist*, 48:384–384, 1993.

[EBDP96] I. Essa, S. Basu, T. Darrell, and A. Pentland. Modeling, tracking and interactive animation of faces and heads using input from video. In *Computer Animation'96. Proceedings*, pages 68–79. IEEE, 1996.

[Ebi04] Y. Ebisawa. Realtime 3D position detection of human pupil. In *Virtual Environments, Human-Computer Interfaces and Measurement Systems, 2004.(VECIMS). 2004 IEEE Symposium on*, pages 8–12, 2004.

[EF77] P. Ekman and W.V. Friesen. *Facial action coding system*. Consulting Psychologists Press, Stanford University, Palo Alto, 1977.

[EHRW98] A. Eleftheriadis, C. Herpel, G. Rajan, and L. Ward. Mpeg-4 systems, text for iso/iec fcd 14496-1 systems. Technical report, MPEG-4 SNHC, 1998.

[FH05] P.F. Felzenszwalb and D.P. Huttenlocher. Pictorial structures for object recognition. *International Journal of Computer Vision*, 61(1):55–79, 2005.

[FKY08] W. Feng, B. Kim, and Y. Yu. Real-time data driven deformation using kernel canonical correlation analysis. *ACM Transactions on Graphics (TOG)*, 27(3):91, 2008.

[FSRE07] J.R.J. Fontaine, K.R. Scherer, E.B. Roesch, and P.C. Ellsworth. The world of emotions is not two-dimensional. *Psychological science*, 18(12):1050–1057, 2007.

[GBGB01] K. Grauman, M. Betke, J. Gips, and G.R. Bradski. Communication via eye blinks-detection and duration analysis in real time. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–1010, 2001.

[GBL+03] K. Grauman, M. Betke, J. Lombardi, J. Gips, and G.R. Bradski. Communication via eye blinks and eyebrow raises: Video-based human-computer interfaces. *Universal Access in the Information Society*, 2(4):359–373, 2003.

[GCJB03] A.B.V. Graciano, R.M. Cesar Jr, and I. Bloch. Inexact graph matching for facial feature segmentation and recognition in video sequences: Results on face tracking. In *Progress in Pattern Recognition, Speech and Image Analysis*, pages 71–78. Springer, 2003.

[GKRR01] R. Goldenberg, R. Kimmel, E. Rivlin, and M. Rudzsky. Fast geodesic active contours. *Image Processing, IEEE Transactions on*, 10(10):1467–1475, 2001.

[GL10] E.R. Gast and M.S. Lew. A framework for real-time face and facial feature tracking using optical flow pre-estimation and template tracking. Master's thesis, LIACS, Leiden University, 2010.

[GM98] C.A. Glasbey and K.V. Mardia. A review of image-warping methods. *Journal of applied statistics*, 25(2):155–171, 1998.

[GMDlTM+07] J. Gonzalez-Mora, F. De la Torre, R. Murthi, N. Guil, and E.L. Zapata. Bilinear active appearance models. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE, 2007.

[GSLT09] X. Gao, Y. Su, X. Li, and D. Tao. Gabor texture in active appearance models. *Neurocomputing*, 72(13):3174–3181, 2009.

[GSLT10] X. Gao, Y. Su, X. Li, and D. Tao. A review of active appearance models. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 40(2):145–158, 2010.

[HAS11] J.P. Hansen, J.S. Augustin, and H. Skovsgaard. Gaze interaction from bed. In *Proceedings of the 1st Conference on Novel Gaze-Controlled Applications*, page 11, 2011.

[Her] Hercules. Hercules webcam. http://www.hercules.com/fr/webcam/.

[HFDL06] A.C. Hodge, A. Fenster, D.B. Downey, and H.M. Ladak. Prostate boundary segmentation from ultrasound images using 2d active shape models: Optimisation and extension to 3d. *Computer methods and programs in biomedicine*, 84(2-3):99–113, 2006.

[HFR+11] Y. Hou, P. Fan, I. Ravyse, V. Enescu, and H. Sahli. Smooth adaptive fitting of 3d face model for the estimation of rigid and nonrigid facial motion in video sequences. *Signal Processing: Image Communication*, 26(8):550–566, 2011.

[HJ10] D.W. Hansen and Q. Ji. In the eye of the beholder: A survey of models for eyes and gaze. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(3):478–500, 2010.

[HNH+02] D.W. Hansen, M. Nielsen, J.P. Hansen, A.S. Johansen, and M.B. Stegmann. Tracking eyes using shape and appearance. In *IAPR Workshop on Machine Vision Applications-MVA*, pages 201–204, 2002.

[HP03] D.W. Hansen and A.E.C. Pece. Iris tracking with feature free contours. In *Analysis and Modeling of Faces and Gestures, 2003. AMFG 2003. IEEE International Workshop on*, pages 208–214, 2003.

[Hér07] R. Hérault. *Vision et apprentissage statistique pour la reconnaissance d'items*. PhD thesis, Université de Technologie de Compiègne, 2007.

[HSL11] T. Heyman, V. Spruyt, and A. Ledda. 3d face tracking and gaze estimation using a monocular camera. In *Proceedings of the 2nd International Conference on Positioning and Context-Awareness*, pages 1–6, 2011.

[Hsu02] R.L. Hsu. Face detection and modeling for recognition. Technical report, DTIC Document, 2002.

[IBMK04] T. Ishikawa, S. Baker, I. Matthews, and T. Kanade. *Passive driver gaze tracking with active appearance models*. Citeseer, 2004.

[IJSM09] P. Isokoski, M. Joos, O. Spakov, and B. Martin. Gaze controlled games. *Universal Access in the Information Society*, 8(4):323–337, 2009.

[IMM] IMMEMO. (immersion 3d basée sur l'interaction Émotionnelle). http://www.rennes.supelec.fr/immemo/.

[Iva07] P. Ivan. *Active appearance models for gaze estimation*. PhD thesis, Vrije University, Amsterdam, 2007.

[Jol05] I. Jolliffe. *Principal component analysis*. Wiley Online Library, 2005.

[JTDP05] P. Joshi, W.C. Tien, M. Desbrun, and F. Pighin. Learning controls for blend shape based realistic facial animation. In *ACM SIGGRAPH 2005 Courses*, page 8. ACM, 2005.

[Kal93] P. Kalra. *An Interactive Multimodal Facial Animation System*. PhD thesis, Ecole Polytechnique Fédérale de Lausanne, Switzerland, 1993.

[KCT00] Takeo Kanade, Jeffrey F Cohn, and Yingli Tian. Comprehensive database for facial expression analysis. In *Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference on*, pages 46–53. IEEE, 2000.

[KHH05] P. Kuo, P. Hillman, and J. Hannah. Improved facial feature extraction for model-based multimedia. In *Proceedings 2nd IEE European Conference on Visual Media Production*, pages 137–146. Citeseer, 2005.

[Khi10] R. Khilari. Iris tracking and blink detection for human-computer interaction using a low resolution webcam. In *Proceedings of the Seventh Indian Conference on Computer Vision, Graphics and Image Processing*, ICVGIP '10, pages 456–463, New York, NY, USA, 2010.

[KK07] J.T. Kim and D. Kim. Gaze tracking with active appearance models. In *Proceeding of The 7th POSTECH-KYUTECH Joint Workshop On Neuroinfomatics*, pages 90–92, 2007.

[KMTT92] P. Kalra, A. Mangili, N.M Thalmann, and D. Thalmann. Simulation of facial muscle actions based on rational free form deformations. *Computer Graphics Forum*, 11(3):59–69, 1992.

[KR03] T. Kawaguchi and M. Rizon. Iris detection using intensity and edge information. *Pattern Recognition*, 36(2):549–562, 2003.

[KS06] F. Kahraman and M.B. Stegmann. Towards illumination-invariant localization of faces using active appearance models. In *Signal Processing Symposium, 2006. NORSIG 2006. Proceedings of the 7th Nordic*, pages 102–105. IEEE, 2006.

[KS11] V. Kazemi and J. Sullivan. Face alignment with part-based modeling. In *Proceedings of the British Machine Vision Conference*, pages 27–1. BMVA Press, 2011.

[KWT88] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International journal of computer vision*, 1(4):321–331, 1988.

[LCIS98] M. La Cascia, J. Isidoro, and S. Sclaroff. Head tracking via robust registration in texture map images. In *Computer Vision and Pattern Recognition, 1998. Proceedings. 1998 IEEE Computer Society Conference on*, pages 508–514. IEEE, 1998.

[LI05] Y. Li and W. Ito. Shape parameter optimization for adaboosted active shape model. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 1, pages 251–258. IEEE, 2005.

[Lim10] W.S.P. Lima. Face recognition using 3d structural geometry of rigid features extracted from 2d images. Master's thesis, Universidade do Minho, Escola de Engenharia, 2010.

[LK81] B.D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *International Joint Conference on Artificial Intelligence*, pages 674–679, 1981.

[LM08] B. Lance and S.C. Marsella. The relation between gaze behavior and the attribution of emotion: An empirical study. In *Intelligent Virtual Agents*, pages 1–14. Springer, 2008.

[LPDB05] G. Langs, P. Peloschek, R. Donner, and H. Bischof. A clique of active appearance models by minimum description length. In *British Machine Vision Conference (BMCV'05)*, pages 859–868, 2005.

[LTP08] J.S. Lobmaier, B.P. Tiddeman, and D.I. Perrett. Emotional expression modulates perceived gaze direction. *Emotion*, 8(4):573–577, 2008.

[MA07] M. Meyer and J. Anderson. Key point subspace acceleration and soft caching. *ACM Transactions on Graphics (TOG)*, 26(3):74, 2007.

[MB04] I. Matthews and S. Baker. Active appearance models revisited. *International Journal of Computer Vision*, 60(2):135–164, 2004.

[MCRH06] D. Marshall, D. Cosker, P.L. Rosin, and Y. Hicks. Speech and expression driven animation of a video-realistic appearance based hierarchical facial model. In *Workshop in conjunction with IEEE CVPR of Learning, Representation and Context for Human Sensing in Video*. Citeseer, 2006.

[MKXC06] T. Moriyama, T. Kanade, J. Xiao, and J. F Cohn. Meticulously detailed eye region model and its application to analysis of facial images. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELIGENCE*, 28(5):738–752, 2006.

[MMKC08] P. Maurel, A. McGonigal, R. Keriven, and P. Chauvel. 3d model fitting for facial expression analysis under uncontrolled imaging conditions. In *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, pages 1–4, 2008.

[MN08] S. Milborrow and F. Nicolls. Locating facial features with an extended active shape model. In *Computer Vision–ECCV 2008*, pages 504–513. Springer, 2008.

[MP00] M. Malciu and F.J. Preteux. Tracking facial features in video sequences using a deformable-model-based approach. In *International Symposium on Optical Science and Technology*, pages 51–62. International Society for Optics and Photonics, 2000.

[MR11] C. Mayer and B. Radig. Learning displacement experts from multi-band images for face model fitting. In *ACHI 2011, The Fourth International Conference on Advances in Computer-Human Interactions*, pages 106–111, 2011.

[MSMM90] S. Menet, P. Saint-Marc, and G. Medioni. Active contour models: Overview, implementation and applications. In *Systems, Man and Cybernetics, 1990. Conference Proceedings., IEEE International Conference on*, pages 194–199. IEEE, 1990.

[MTPT88] N. Magnenat-Thalmann, E. Primeau, and D. Thalmann. Abstract muscle action procedures for human face animation. *The Visual Computer*, 3(5):290–297, 1988.

[MVCP10] G. McKeown, M.F. Valstar, R. Cowie, and M. Pantic. The SEMAINE corpus of emotionally coloured character interactions. In *Multimedia and Expo (ICME), 2010 IEEE International Conference on*, 2010.

[MVdM96] T. Maurer and C. Von der Malsburg. Tracking and learning graphs and pose on image sequences of faces. In *Automatic Face and Gesture Recogni-*

tion, 1996., Proceedings of the Second International Conference on, pages 176–181. IEEE, 1996.

[MW67] A. Mehrabian and M. Wiener. Decoding of inconsistent communications. *Journal of Personality and Social Psychology*, 6(1):109–114, 1967.

[NM64] J.A. Nelder and R. Mead. A simplex method for function minimization. *Computer Journal*, 7:308–313, 1964.

[NRB+12] J. Nicolle, V. Rapp, K. Bailly, L. Prevost, and M. Chetouani. Robust continuous prediction of human emotions using multiscale dynamic cues. In *Proceedings of the 14th ACM international conference on Multimodal interaction*, pages 501–508. ACM, 2012.

[Opt] Optitrack. Optitrack infrared webcam. http://www.naturalpoint.com/optitrack/hardware/.

[Oro07] J. Orozco. *face detection and tracking for facial expression analysis*. PhD thesis, Universitat Autónoma de Barcelona, 2007.

[OSM12] D. Ozkan, S. Scherer, and L. Morency. Step-wise emotion recognition using concatenated-hmm. In *Proceedings of the 14th ACM international conference on Multimodal interaction*, pages 477–484. ACM, 2012.

[P.97] Ramani P. Snakes: an active model, july 1997.

[Par74] F. I. Parke. *A Parametric Model for Human Faces*. PhD thesis, University of Utah, 1974.

[Par82] F.I. Parke. Parameterized models for facial animation. *Computer Graphics and Applications, IEEE*, 2(9):61–68, 1982.

[PB81] S.M. Platt and N.I. Badler. Animating facial expressions. *ACM SIGGRAPH computer graphics*, 15(3):245–252, 1981.

[PBMD07] J. Peyras, A. Bartoli, H. Mercier, and P. Dalle. Segmented aams improve person-independent face fitting. In *In BMVC'07-Proceedings of the 18th British Machine Vision Conference*. Citeseer, 2007.

[PCG+03] A. Perez, M.L. Cordoba, A. Garcia, R. Mendez, M.L. Munoz, J.L. Pedraza, and F. Sanchez. A precise Eye-Gaze detection and tracking system. In *WSCG*, 2003.

[PCMC01] X.M. Pardo, M.J. Carreira, A. Mosquera, and D. Cabello. A snake for ct image segmentation integrating region and edge information. *Image and Vision Computing*, 19(7):461–475, 2001.

[PHL+06] F. Pighin, J. Hecker, D. Lischinski, R. Szeliski, and D.H. Salesin. Synthesizing realistic facial expressions from photographs. In *ACM SIGGRAPH 2006 Courses*, page 19. ACM, 2006.

[PM08] G. Papandreou and P. Maragos. Adaptive and constrained algorithms for inverse compositional active appearance model fitting. In *Computer*

Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, pages 1–8. IEEE, 2008.

[PS09] A. Patel and W.A.P. Smith. 3d morphable face models revisited. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1327–1334. IEEE, 2009.

[PSWL07] G. Pan, L. Sun, Z. Wu, and S. Lao. Eyeblink-based anti-spoofing in face recognition from a generic webcamera. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE, 2007.

[PWL05] K.R. Park, M.C. Whang, and J.S. Lim. A study on non-intrusive facial and eye gaze detection. In *Advanced Concepts for Intelligent Vision Systems*, pages 52–59. Springer, 2005.

[QX02] J. Qiang and Y. Xiaojie. Real-time eye, gaze, and face pose tracking for monitoring driver vigilance. *Real-Time Imaging*, 8(5):357–377, 2002.

[RBCG08] A. Rakotomamonjy, F. Bach, S. Canu, and Y. Grandvalet. Simplemkl. *Journal of Machine Learning Research*, 9:2491–2521, 2008.

[RCA03] M.G. Roberts, T.F. Cootes, and J.E. Adams. Linking sequences of active appearance sub-models via constraints: an application in automated vertebral morphometry. In *14th British Machine Vision Conference*, volume 1, pages 349–358, 2003.

[RCY+11] M.J. Reale, S. Canavan, L. Yin, K. Hu, and T. Hung. A Multi-Gesture interaction system using a 3-D iris disk model for gaze estimation and an active appearance model for 3-D hand pointing. *Multimedia, IEEE Transactions on*, 13(3):474–486, 2011.

[Rec71] I. Rechenberg. *Evolutionsstrategie – Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*. PhD thesis, Berlin Technical University, 1971.

[RG06] M. Rogers and J. Graham. Robust active shape model search. In *Computer Vision-ECCV 2002*, pages 517–530. Springer, 2006.

[RM95] P. Radeva and E. Martí. Facial features segmentation by model-based snakes. In *International Conference on Computing Analysis and Image Processing, Prague*, pages 1–5, 1995.

[Ros04] A. Ross. Procrustes analysis. Technical report, Department of Computer Science and Engineering, University of South Carolina, 2004.

[RS08] U. Ravyse and H. Sahli. A biomechanical model for image-based estimation of 3d face deformations. In *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, pages 1089–1092. IEEE, 2008.

[RSBP11] V. Rapp, T. Senechal, K. Bailly, and L. Prevost. Multiple kernel learning svm and statistical validation for facial landmark detection. In *IEEE Int'l. Conf. Face and Gesture Recognition (FG'11)*, pages 265–271, 2011.

[RWDB08] W.J. Ryan, D.L. Woodard, A.T. Duchowski, and S.T. Birchfield. Adapting starburst for elliptical iris segmentation. In *IEEE second international conference on biometrics : Theory, Applications ans Systems*, Washington D.C., September 2008.

[Ryd87] M. Rydfalk. Candide, a parameterized face. Technical report, Report No LiTH-ISY-I-866, Dept. of Electrical Engineering, Linkoping University, Sweden, 1987.

[SAD+08a] A. Savran, N. Aly\üz, H. Dibeklioğlu, O. Çeliktutan, B. G\ökberk, B. Sankur, and L. Akarun. Bosphorus database for 3D face analysis. *Biometrics and Identity Management*, page 47–56, 2008.

[SAD+08b] A. Savran, N. Aly\üz, H. Dibeklioğlu, O. Çeliktutan, B. G\ökberk, B. Sankur, and L. Akarun. Bosphorus database for 3D face analysis. *Biometrics and Identity Management*, pages 47–56, 2008.

[SALGS07a] A. Sattar, Y. Aidarous, S. Le Gallou, and R. Seguier. Face alignment by 2.5 d active appearance model optimized by simplex. *ICVS, Bielefeld University, Germany*, 2007.

[SALGS07b] A. Sattar, Y. Aidarous, S. Le Gallou, and R. Seguier. Face alignment by 2.5d active appearance model optimized by simplex. In *International Conference on Computer Vision Systems (ICVS)*, pages 1–10, 2007.

[SAS08] A. Sattar, Y. Aidarous, and R. Seguier. Gagm-aam: a genetic optimization with gaussian mixtures for active appearance models. In *IEEE International Conference on Image Processing (ICIP'08)*, pages 3220–3223, 2008.

[SBS10] N. Stoiber, G. Breton, and R. Seguier. Modeling short-term dynamics and variability for realistic interactive facial animation. *Computer Graphics and Applications, IEEE*, 30(4):51–61, 2010.

[SCGH05] N. Sebe, I. Cohen, T. Gevers, and T.S. Huang. Multimodal approaches for emotion recognition: a survey. In *Electronic Imaging 2005*, pages 56–67. International Society for Optics and Photonics, 2005.

[SCS+12] A. Savran, H. Cao, M. Shah, A. Nenkova, and R. Verma. Combining video, audio and lexical indicators of affect in spontaneous conversation via particle filtering. In *Proceedings of the 14th ACM international conference on Multimodal interaction*, pages 485–492, 2012.

[SG07] J. Saragih and R. Goecke. A nonlinear discriminative approach to aam fitting. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE, 2007.

[SGTL08] Y. Su, X. Gao, D. Tao, and X. Li. Gabor-based texture representation in aams. In *IEEE International Conference on Systems, Man and Cybernetics*, pages 2236–2240. IEEE, 2008.

[Shl05] J. Shlens. A tutorial on principal component analysis. Technical report, Systems Neurobiology Laboratory, University of California at San Diego, 2005.

[SI98] S. Sclaroff and J. Isidoro. Active blobs. In *Computer Vision, 1998. Sixth International Conference on*, pages 1146–1153, 1998.

[SK07] J. Sung and D. Kim. A background robust active appearance model using active contour technique. *Pattern recognition*, 40(1):108–120, 2007.

[SKK07] J. Sung, T. Kanade, and D. Kim. A unified gradient-based approach for combining asm into aam. *International Journal of Computer Vision*, 75(2):297–309, 2007.

[SKM04] J. Short, J. Kittler, and K. Messer. A comparison of photometric normalisation algorithms for face verification. In *IEEE Automatic Face and Gesture Recognition (AFGR)*, pages 1–6, 2004.

[SL09] H.J. Shin and Y. Lee. Expression synthesis and transfer in parameter spaces. *Computer Graphics Forum*, 28(7):1829–1835, 2009.

[SLC11] J.M. Saragih, S. Lucey, and J.F. Cohn. Deformable model fitting by regularized landmark mean-shift. *International Journal of Computer Vision*, 91(2):200–215, 2011.

[SLGBG09] R. Séguier, S. Le Gallou, G. Breton, and C. Garcia. Adapted active appearance models. *EURASIP Journal on Image and Video Processing*, 2009(10), 2009.

[SLPP12] M. Soleymani, J. Lichtenauer, T. Pun, and M. Pantic. A multimodal database for affect recognition and implicit tagging. *Affective Computing, IEEE Transactions on*, 3(1):42–55, 2012.

[SLS+07] N. Sebe, M.S. Lew, Y. Sun, I. Cohen, T. Gevers, and T.S. Huang. Authentic facial expression analysis. *Image and Vision Computing*, 25(12):1856–1863, 2007.

[SMG09] R. Stricker, C. Martin, and H. Gross. Increasing the robustness of 2d active appearance models for real-world applications. In *International Conference on Computer Vision Systems*, pages 364–373. Springer, 2009.

[SNSM12] S. Sidra Naveed, B. Sikander, and Khiyal M.S.H. Eye tracking system with blink detection. *Journal Of Computing*, 4, 2012.

[SRD02] S. Sirohey, A. Rosenfeld, and Z. Duric. A method of detecting and tracking irises and eyelids in video. *Pattern Recognition*, 35(6):1389–1401, 2002.

[SRS+11] Thibaud Senechal, Vincent Rapp, Hanan Salam, Renaud Seguier, Kevin Bailly, and L. Prevost. Combining aam coefficients with lgbp histograms in the multi-kernel svm framework to detect facial action units. In *Automatic Face & Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on*, pages 860–865. IEEE, 2011.

[SRS+12] T Sénéchal, V. Rapp, H. Salam, R. Seguier, K. Bailly, and L. Prevost. Facial action recognition combining heterogeneous features via multi-kernel learning. *IEEE Transactions on Systems, Man, and Cybernetics–Part B*, 42(4):993–1005, 2012.

[SS86] T. W. Sederberg and P. R. Scott. Free-form deformation of solid geometric models. *ACM SIGGRAPH computer graphics*, 20(4):151–160, 1986.

[SS10] A. Sattar and R. Seguier. Facial feature extraction using hybrid genetic-simplex optimization in multi-objective active appearance model. In *Digital Information Management (ICDIM), 2010 Fifth International Conference on*, pages 152–158. IEEE, 2010.

[SSS12] C. Soladie, N. Stoiber, and R. Seguier. A new invariant representation of facial expressions: Definition and application to blended expression recognition. In *Proceedings of the 2012 IEEE International Conference on Image Processing*, 2012.

[SSSS13] C. Soladié, H. Salam, N. Stoiber, and R. Séguier. Continuous facial expression representation for multimodal emotion detection. *International Journal of Advanced Computer Science (IJACSci)*, 3(5), 2013.

[Sto10] N. Stoiber. *Modeling Emotional Facial Expressions and their Dynamics for Realistic Interactive Facial Animation on Virtual Characters*. PhD thesis, Université de Rennes1, 2010.

[SUB09] M. Storer, M. Urschler, and H. Bischof. 3d-mam: 3d morphable appearance model for efficient fine head pose estimation from still images. In *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*, pages 192–199. IEEE, 2009.

[SVCP12] B. Schuller, M. Valstar, R. Cowie, and M. Pantic. Avec 2012–the continuous audio/visual emotion challenge. In *Proceedings 2nd International Audio/Visual Emotion Challenge and Workshop, AVEC*, pages 449–456, 2012.

[SY97] R. Stiefelhagen and J. Yang. Gaze tracking for multimodal human-computer interaction. In *icassp*, page 2617, 1997.

[TAiMZP12] G. Tzimiropoulos, J. Alabort-i Medina, S. Zafeiriou, and M. Pantic. Generic active appearance models revisited. In *Computer Vision–ACCV 2012*, pages 650–663. Springer, 2012.

[TBA+09] P. Tresadern, H. Bhaskar, S. Adeshina, C.J. Taylor, and T.F. Cootes. Combining local and global shape models for deformable object matching. In *Proc. British Machine Vision Conference*, pages 1–12, 2009.

[TCMH11] L.C. Trutoiu, E.J. Carter, I. Matthews, and J.K. Hodgins. Modeling and animating eye blinks. *ACM Transactions on Applied Perception (TAP)*, 8(3):17, 2011.

[TH99] H. Tao and T.S. Huang. Explanation-based facial motion tracking using a piecewise bezier volume deformation model. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, pages 1–7. IEEE, 1999.

[TKC00] Y. Tian, T. Kanade, and Jeffrey F. Cohn. Dual-state parametric eye tracking. In *Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference on*, pages 110–115. IEEE, 2000.

[TP91] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of cognitive neuroscience*, 3(1):71–86, 1991.

[TW90] D. Terzopoulos and K. Waters. Physically-based facial modelling, analysis, and animation. *The journal of visualization and computer animation*, 1(2):73–80, 1990.

[VdKS11] J. Van der Kamp and V. Sundstedt. Gaze and voice controlled drawing. In *Novel Gaze-Controlled Applications (NGCA)*, page 9, 2011.

[Ver99] R. Vertegaal. The gaze groupware system: mediating joint attention in multiparty communication and collaboration. In *Proceedings of the SIGCHI conference on Human factors in computing systems: the CHI is the limit*, pages 294–301. ACM, 1999.

[VG08] R. Valenti and T. Gevers. Accurate eye center location and tracking using isophote curvature. In *CVPR*, pages 1–8, 2008.

[VJ04] P. Viola and M.J. Jones. Robust real-time face detection. *International journal of computer vision*, 57(2):137–154, 2004.

[VJM+11] M.F. Valstar, B. Jiang, M. Mehu, M. Pantic, and K. Scherer. The first facial expression recognition and analysis challenge. In *Automatic Face & Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on*, pages 921–926. IEEE, 2011.

[VP06] M. Valstar and M. Pantic. Fully automatic facial action unit detection and temporal analysis. In *Computer Vision and Pattern Recognition Workshop, 2006. CVPRW'06. Conference on*, pages 149–149. IEEE, 2006.

[VSG12] R. Valenti, N. Sebe, and T. Gevers. Using geometric properties of topographic manifold to detect and track eyes for human-computer interaction. *IEEE Transactions on Image Processing*, 21(2):802–815, 2012.

[Wat87] K. Waters. A muscle model for animation three-dimensional facial expression. *ACM SIGGRAPH Computer Graphics*, 21(4):17–24, 1987.

[WBR+11] T. Wu, N.J. Butko, P. Ruvolo, J. Whitehill, M.S. Bartlett, and J.R. Movellan. Action unit recognition transfer across datasets. In *Automatic Face & Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on*, pages 889–896. IEEE, 2011.

[WFK97] L. Wiskott, J.M. Fellous, and C. Kuiger, N .and Von der Malsburg. Face recognition by elastic bunch graph matching. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 19(7):775–779, 1997.

[WH04] Z. Wen and T. S. Huang. *3D face processing, modeling, analysis and synthesis*. Kluwer academic publishers, 2004.

[WKW+07] H. Wu, Y. Kitagawa, T. Wada, T. Kato, and Q. Chen. Tracking iris contour with a 3D eye-model for gaze estimation. In *Proceedings of the 8th Asian conference on Computer vision-Volume Part I*, pages 688–697, 2007.

[WLSN07] U. Weidenbacher, G. Layher, P.M. Strauss, and H. Neumann. A comprehensive head pose and gaze database. In *3rd IET International Conference on Intelligent Environments (IE 07)*, pages 455–458, 2007.

[WLZ04] Y. Wu, H. Liu, and H. Zha. A new method of detecting human eyelids based on deformable templates. In *Systems, Man and Cybernetics, 2004 IEEE International Conference on*, volume 1, pages 604–609. IEEE, 2004.

[WULO06] A. Weissenfeld, O. Urfalioglu, K. Liu, and J. Ostermann. Robust rigid head motion estimation based on differential evolution. In *Multimedia and Expo, 2006 IEEE International Conference on*, pages 225–228, 2006.

[WVS+06] T. Whitmarsh, R.C. Veltkamp, M. Spagnuolo, S. Marini, and F.B. ter Haar. Landmark detection on 3d face scans by facial model registration. In *1st international symposium on shapes and semantics*, pages 71–75, 2006.

[XBMK04] J. Xiao, S. Baker, I. Matthews, and T. Kanade. Real-time combined 2d+3d active appearance models. In *Proceedings of the 2004 IEEE computer society conference on Computer vision and pattern recognition*, pages 535–542, 2004.

[XCZL08] Z. Xu, H. Chen, S. Zhu, and J. Luo. A hierarchical compositional model for face representation and sketching. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(6):955–969, 2008.

[XWLZ09] Guoqing X., Yangsheng W., Jituo L., and Xiaoxu Z. Real time detection of eye corners and iris center from images acquired by usual camera. *Intelligent Networks and Intelligent Systems, International Workshop on*, 0:401–404, 2009.

[XY09] W. Xin and T. Yunxia. A faster b spline snake. In *Robotics and Biomimetics (ROBIO), 2009 IEEE International Conference on*, pages 2314–2319. IEEE, 2009.

[YB11] S. Yang and B. Bhanu. Facial expression recognition using emotion avatar image. In *Automatic Face & Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on*, pages 866–871. IEEE, 2011.

[YDJ+13] G. Yongxin, Y. Dan, L. Jiwen, L. Bo, and Z. Xiaohong. Active appearance models using statistical characteristics of gabor based texture representation. *Journal of Visual Communication and Image Representation*, 24(5):627–634, 2013.

[YHC92a] A.L. Yuille, P. W Hallinan, and David S Cohen. Feature extraction from faces using deformable templates. *International journal of computer vision*, 8(2):99–111, 1992.

[YHC92b] A.L. Yuille, P.W. Hallinan, and D.S. Cohen. Feature extraction from faces using deformable templates. *International journal of computer vision*, 8(2):99–111, 1992.

[YUYA08] H. Yamazoe, A. Utsumi, T. Yonezawa, and S. Abe. Remote and Head-Motion-Free gaze tracking for real environments with automated Head-Eye model calibrations. In *IEEE CVPR Workshop human communicative behavior analysis*, pages 1–6, June 2008.

[YXTK10] P. Yan, S. Xu, B. Turkbey, and J. Kruecker. Discrete deformable model guided by partial active shape model for trus image segmentation. *Biomedical Engineering, IEEE Transactions on*, 57(5):1158–1166, 2010.

[YZ07] J.A. Ybanez Zepeda. *A linear estimation of the face's tridimensional pose and facial expressions*. PhD thesis, Telecom ParisTech, 2007.

[ZC05] C. Zhang and F.S. Cohen. Component-based active appearance models for face modelling. In *Advances in Biometrics*, pages 206–212. Springer, 2005.

[ZG05] L. Zalewski and S. Gong. 2d statistical models of facial expressions for realistic 3d avatar animation. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 217–222. IEEE, 2005.

[ZGZ03] Y. Zhou, L. Gu, and H.J. Zhang. Bayesian tangent shape model: Estimating shape and pose parameters via bayesian inference. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 1, pages I–109. IEEE, 2003.

[ZJ04] Z. Zhu and Q. Ji. Eye and gaze tracking for interactive graphic display. *Machine Vision and Applications*, 15(3):139–148, 2004.

[ZJ05a] Y. Zhang and Q. Ji. Active and dynamic information fusion for facial expression understanding from image sequences. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(5):699–714, 2005.

[ZJ05b] Z. Zhu and Q. Ji. Robust real-time eye detection and tracking under variable lighting conditions and various face orientations. *Computer Vision and Image Understanding*, 98(1):124–154, 2005.

[ZPRH09] Z. Zeng, M. Pantic, G.I. Roisman, and T.S. Huang. A survey of affect recognition methods: Audio, visual, and spontaneous expressions. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(1):39–58, 2009.

[ZR12] X. Zhu and D. Ramanan. Face detection, pose estimation and landmark localization in the wild. In *Computer Vision and Pattern Recognition (CVPR)*, pages 68–79, 2012.

[ZSG+05] W. Zhang, S. Shan, W. Gao, X. Chen, and H. Zhang. Local gabor binary pattern histogram sequence (lgbphs): A novel non-statistical model for face representation and recognition. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 1, pages 786–791. IEEE, 2005.